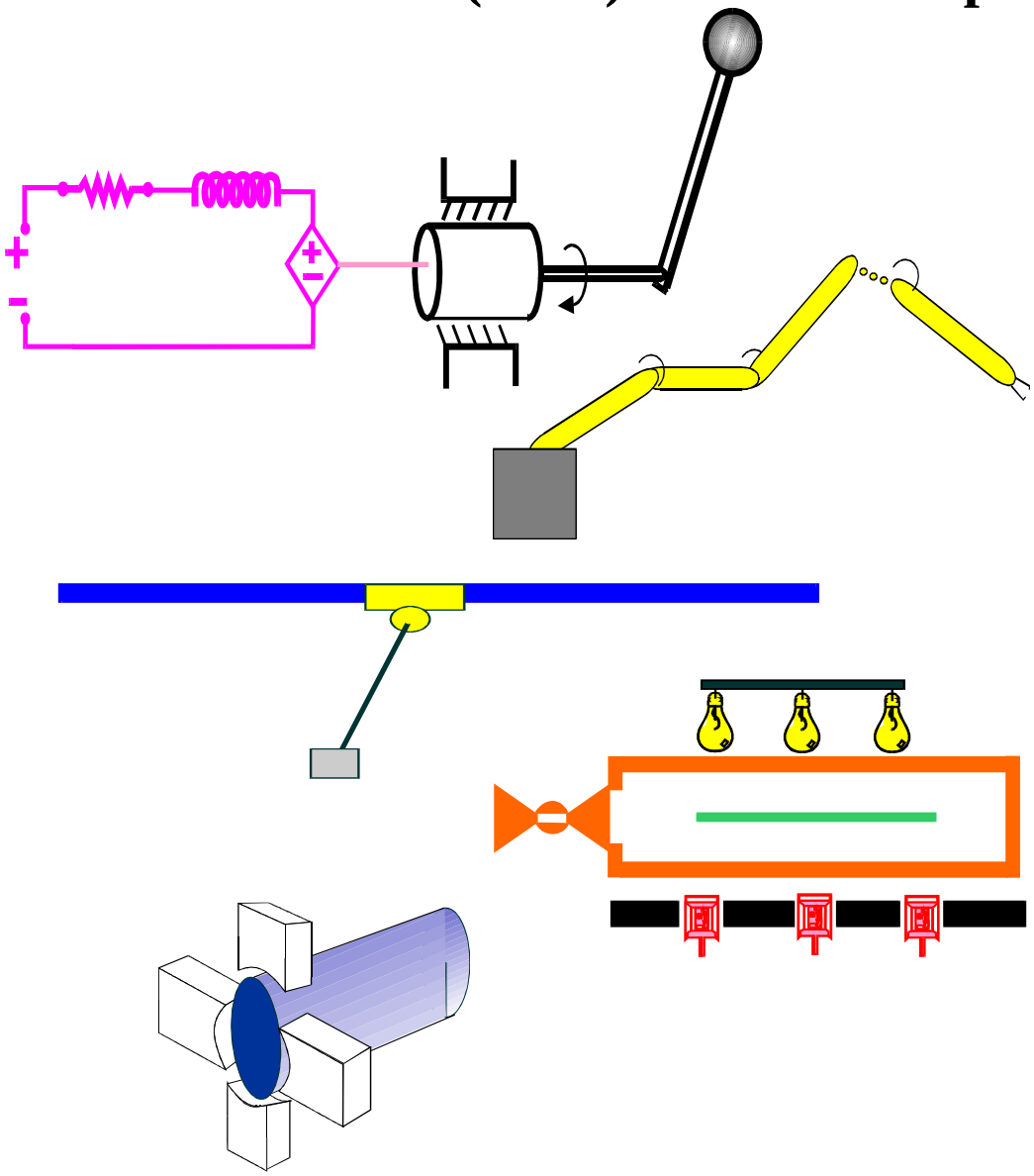
Clemson University
College of Engineering and Science
Control and Robotics (CRB) Technical Report



Number: CU/CRB/3/11/07/#1

Title: Control of a Remotely Operated Quadrotor Aerial Vehicle and Camera Unit Using a Fly-The-Camera Perspective

Authors: DongBin Lee, Vilas Chitrakaran, Timothy C. Burg, Darren M. Dawson, and Bin Xian

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
11 MAR 2007		**00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Control of a Remotely Operated Quadrotor Aerial Vehicle and Camera Unit Using a Fly-The-Camera Perspective	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Clemson University ,College of Engineering and Science,109 Riggs Hall,Clemson,SC,29631-0901**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**15**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

Control of a Remotely Operated Quadrotor Aerial Vehicle and Camera Unit Using a Fly-The-Camera Perspective

DongBin Lee[1], Vilas Chitrakaran[2], Timothy Burg[1], Darren Dawson[1], and Bin Xian[3]

Abstract— This paper presents a mission-centric approach to controlling the optical axis of a video camera mounted on a camera manipulator and fixed to a quadrotor remotely operated vehicle. The approach considers that for video collection tasks a single operator should be able to operate the systems by "flying-the-camera"; that is, collect video from the perspective that the operator is looking out of and is the pilot of the camera. This will allow the control of the quadrotor and the camera positioner to be fused into a single control problem where the camera is positioned using the four degree-of-freedom (DOF) quadrotor and the two DOF camera positioner to provide a full six DOF actuation of the camera view. The closed-loop controller is designed based on a Lyapunov-type analysis and is shown to produce Globally Uniformly Ultimately Bounded (GUUB) tracking of a desired trajectory. Computer simulation results are provided to demonstrate the performance of the suggested controller.

I. Introduction

The potential for unmanned aerial vehicles (UAVs) in applications as diverse as fire fighting, emergency response, military and civilian surveillance, crop monitoring, and geographical registration has been well established. Many research groups have provided convincing demonstrations of the utility of UAVs in these applications. However, there is still a large chasm between the anticipated "tool of the future" and currently available systems. The commercial and military use of UAVs is predicated on the ability of such vehicles to perform new, safer, or more cost effective tasks than traditional manned aircraft. Until recently, this has been more of a question than a statement; however, recent advances in aerial vehicle construction, sensors, digital electronics, control design have seen a rapid increase in UAV applications.

Aerial vehicle construction should be considered as an important factor in UAV applications. Improved manufacturing techniques are capable of producing small, complex, precise parts at a reasonable price and new battery technologies have made electric hovering craft more feasible. One of the interesting small aerial vehicles that seems to have benefited from these developments is the quadrotor helicopter depicted in Figure 2. The quad-rotor consists of four independently driven rotating blades that can provide lift in the vertical direction. The vehicle moves in other directions by creating a mismatch between rotor speeds, and hence, this configuration can produce torques about the roll, pitch, and yaw axes. The basic concept for the quad-rotor dates back to 1907; some notes on the history of the quad-rotor and related references can be found in [9]. With this as a backdrop, the focus of the work presented here will be the small quad-rotor family of aerial vehicles. The discussion will be limited to vehicles with less than 0.5kg payload. This weight restriction means that certain technologies that may make sense for larger, more expensive aircraft may not apply to this class of aircraft.

The typical scenario for using the quadrotor helicopter (or any aerial vehicle) as a video camera platform is based on mounting the camera on a positioner that is controlled independently from the vehicle. When the navigation or surveillance tasks become complicated, two people may be required to achieve the camera targeting objective: a pilot to navigate the UAV and a camera operator. It is insightful to consider the actions of these two actors in this scenario in order to hypothesize a new operational mode. The pilot will work to position the aircraft to avoid obstacles and to put the camera platform, i.e., the aerial vehicle, in a position that the will allow the camera operator to watch the camera feed and move the camera positioner to track a target or survey an area. An important underlying action on the part of the camera operator that makes this scenario feasible is that the camera operator must compensate for the motions of the UAV that disturb the camera targeting as illustrated in Figure 1. Additionally, there must be communication between the pilot and the operator so that the camera platform is correctly positioned or moved to meet the video acquisition objective. More specifically, the camera positioning problem is split between the pilot and the camera operator. Since the operator is not in full control of positioning the camera, she must rely on commands to the pilot to provide movement of the camera platform for motions not included in the camera positioner. For example, if the camera positioner is a simple pan-tilt and the camera operator requires translation of the camera then a request must be made to the pilot to move the camera platform. The potential shortcomings of this typical operational scenario can be summarized as: i) multiple skilled technicians are typically required, ii) the camera operator must compensate for the actions of the pilot, and iii) it is not intuitive for a camera operator

[1]Dep't of Electrical and Computer Engineering, Clemson University, Clemson, SC 29634. [2]OC Robotics Ltd., Bristol, BS34 7JU, UK. [2]School of Electrical Engineering and Automation, Tian Jin University, Tian Jin 300072, PR China. email: tburg@clemson.edu

This work is supported in part by a DOC Grant, an ARO Automotive Center Grant, a DOE Contract, a Honda Corporation Grant and a DARPA Contract.

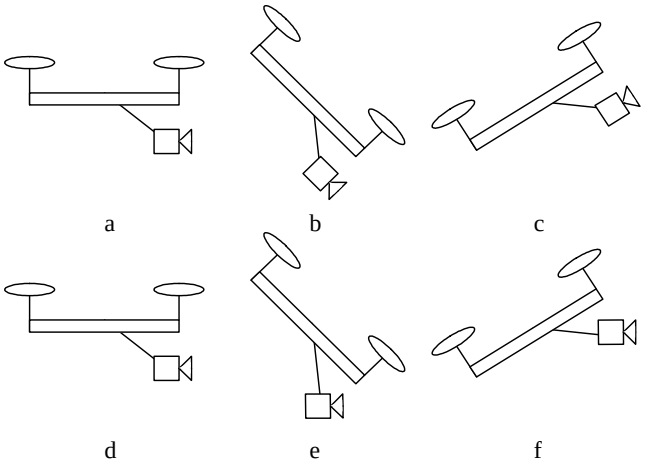


Fig. 1. Diagrams a - c illustrate the effect of uncompensated camera platform motion on the camera axis. Diagrams d - f illustrate a scenario where the camera positioner is used to compenste for the platform motion and maintain the camera view.

to split the camera targeting tasks between actions of the camera positioner controlled by the operator and commands to the pilot.

The problem of providing an intuitive interface with which an operator can move a camera positioner to make a video camera follow a target image appears in many places. The difficulty of moving a system that follows a subject with a video camera was recently addressed in [13] where operating a multilink, redundant-joint camera boom for the movie and television industry is described. The interesting result from this work is that an integrated control strategy, using a vision servoing approach to reduce the number of links controlled by the operator, can improve the use of the system. The final result shows an unexperienced operator achieving the same tracking result as an experienced operator; hence, the control strategy has rendered the system more friendly to the operator. The salient point of the control strategy is that there is independent macro- and micro-positioning of the camera - the operator controls the course positioning and the vision system controls the fine positioning. The authors suggest that the same approach could be used for other camera platforms; however, it is required that the system have redundant positioning axes. Additionally, an automated vision servoing system may not be desirable for general reconnaissance where the target is not known.

A different perspective to this same basic camera targeting problem was presented in [1] and [10] where the camera platform, a quadrotor UAV, and the camera positioning unit are considered to be a single robotic unit. In this work a controller was developed which simultaneously controls both the quadrotor and the camera positioning unit in a complimentary fashion. Both works show combining the four degrees-of-freedom provided by motion of the quadrotor helicopter with two degrees-of-freedom provided by a camera positioner to provide arbitrary six degree-of-freedom positioning of the on-board video camera. The work in [1] is actually directed towards providing an automated means of landing the quadrotor through the vision system but provides an important mathematical framework for the analyzing the combined quadrotor/camera system. The work in [10] builds on [1] to show the design of a velocity controller for the combined quadotor/camera system that works from operator commands generated in the camera field-of-view to move both elements. This perspective, which will be referred to as the fly-the-camera perspective, presents a new interface to the pilot. In this proposed approach, the pilot commands motion from the perspective of the on-board camera - it is as though the pilot is riding on the tip of the camera and commanding movement of the camera ala a six-DOF flying camera. This is subtly different from the traditional remote control approach wherein the pilot processes the camera view and then commands an aircraft motion to create a desired motion of the camera view. The work proposed here exploits this new perspective for fusing vehicle and camera control. In this paper, a quadrotor UAV model will be combined with a two-DOF camera kinematic model to create a fully actuated camera frame and a positioner controller will be designed.

The paper is organized as follows. In Section II, a well known kinematic and dynamic model of the quadrotor is presented. The dynamic model is simplified to include only the translational dynamics of the quadrotor. Assumptions and properties of this model are shown. The kinematics for a three-link camera positioner are developed as a means of studying two special cases that require different two-link positioner configurations: when the camera is looking forward and when the camera is looking downward. The case of the camera looking forward is carried through the control design and simulation while the second case, the camera looking downward, is a simple modification of the first. A Lyapunov function based control design approach is detailed in Section III, its stability analysis is shown in Section IV, and a simulation demonstrating the controller is presented in Section V.

II. System Modeling

A. Underactuated Quadrotor Aerial Vehicle Model

The elements of the quad-rotor unmanned aerial vehicle model are shown in Figure 2 and Figure 3. The quadrotor is assumed to be a rigid body on which thrust and torque act uniformly through the body and that the quadrotor body fixed frame, F, is chosen to coincide with the center of gravity which implies that it has a diagonal inertia matrix. The kinematic and a dynamic model of a quadrotor expressed in the body-fixed reference frame

are as follows [4],[1]

$$\begin{aligned} \dot{p}_{IF}^{I} &= R_F^I(\Theta)\, v_{IF}^F, && (1)\\ \dot{\Theta}_{IF}^{I} &= T_F^I(\Theta)\, \omega_{IF}^F, && (2)\\ \dot{R}_F^I &= R_F^I S\left(\omega_{IF}^F\right), && (3)\\ m\dot{v}_{IF}^F &= F_f^F - mS(\omega_{IF}^F)v_{IF}^F + N_1(v_{IF}^F) + G(R_I^F) && (4) \end{aligned}$$

In this model $v_{IF}^F(t) = [v_x, v_y, v_z]^\intercal \in \mathbb{R}^3$ denotes the linear velocity of the quadrotor body-fixed frame with respect to the earth-fixed inertial frame, I, expressed in the body-fixed frame, F, and $\omega_{IF}^F(t) = [\omega_x, \omega_y, \omega_z]^\intercal \in \mathbb{R}^3$ denotes the angular velocity of the quadrotor body-fixed frame with respect to the inertial frame, I, expressed in the body-fixed frame, F. Equations (1) - (3) represent the kinematics of the quad rotor and include the approximation that the rotational dynamics are negligible. The time derivative of position, $\dot{p}_{IF}^I(t)$ in (1), is the velocity of the quadrotor. In a similar manner, the angular derivative $\dot{\Theta}_{IF}^I(t)$ in (2) represents the angular velocity $\omega_{IF}^F(t)$ transformed by the matrix $T_F^I(\Theta)$. Equation (2) represents the modeling assumption that angular velocity of the quadrotor is calculated directly in lieu of modeling the angular dynamics; that is, $\omega_{IF}^F(t)$ is considered as the system input. The dynamics of the translational velocity is shown in (4) and contains the gravitational term, $G(R_F^I)$, which is represented in the body-fixed frame as

$$G(R_F^I) = mg(R_F^I)^\intercal E_3 \in \mathbb{R}^3 \quad (5)$$

where $g \in \mathbb{R}^1$ denotes gravitational acceleration, $E_3 = [0, 0, 1]^\intercal$ denotes the unit vector in the coordinates of the inertial frame, $m \in \mathbb{R}^1$ is the known mass of the quad-rotor, $N_1\left(v_{IF}^I\right) \in \mathbb{R}^3$ represents a bounded, unknown, nonlinear function (i.e, aerodynamic damping) and $S(\cdot) \in \mathbb{R}^{3\times3}$ is a general form of the skew-symmetric matrix as follows

$$S(\xi) = \begin{bmatrix} 0 & -\xi_3 & \xi_2 \\ \xi_3 & 0 & -\xi_1 \\ -\xi_2 & \xi_1 & 0 \end{bmatrix}, \ \xi = [\xi_1, \xi_2, \xi_3]^T \in \mathbb{R}^3. \quad (6)$$

The quadrotor has inherently six degrees-of-freedom as shown in Figure 2: three translations in the x, y, and z directions and three rotations about the roll (ϕ), pitch (θ), and yaw (ψ) axes. However, the quadrotor has only four control inputs: one translational force along the z-axis and three angular velocities. The vector $F_f^F(t) \in \mathbb{R}^3$ refers to the quadrotor translational forces expressed in the quadrotor frame but in reality represents the single translational force which is created by summing the forces generated by the four rotors and is expressed as

$$F_f^F = B_1 u_1 = \begin{bmatrix} 0 & 0 & u_1 \end{bmatrix}^\intercal \quad (7)$$

where $u_1(t) \in \mathbb{R}^1$.

B. Camera Positioner Kinematics

To create a general analytic framework for modeling various mounting configurations of a two-link revolute positioner a 3-link robot manipulator model is proposed.

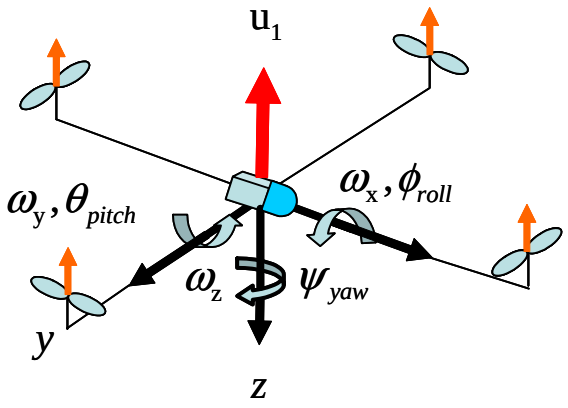


Fig. 2. An underactuated quadrotor vehicle

The camera positioner will be used to augment the camera position with two additional degrees-of-freedom (compared to a camera fixed directly to the quadrotor frame). This general three-axis camera positioner will provide for a Tilt-Pan-Roll motion and the variables $\theta_{tilt}(t)$, $\theta_{pan}(t)$, and $\theta_{roll}(t)$ will be used to represent the Tilt, Pan, and Roll angles, respectively. The Tilt-Pan-Roll motion can be reduced to two special operational cases of Tilt-Roll and Pan-Tilt by freezing (holding constant) one of the joints as shown in Figure 3. The Tilt-Roll configuration can be used to compensate for the quadrotor body roll and pitch when the camera is facing forward for tasks such as general navigation or surveillance. In the Pan-Tilt mode, the camera positioner is also used to compensate for quadrotor body roll and pitch while the camera is facing downward for landing or surveillance tasks. For both of these configurations, the camera optical frame, which is actuated by the combination of the quadrotor and the camera positioner, is fully actuated. The dynamics of the camera unit will be considered negligible and that the angles of the camera positioner can be commanded directly without error.

In order to facilitate the development of the rotation matrices for the manipulator in Figure 3, standard coordinate system definitions are made. Specifically, O_0 is used to represent the origin of a coordinate system at the base of the camera and coincident with camera positioner base coordinate system B and O_3 is the origin of a coordinate system attached at the third link and coincident with camera frame denoted C. Note that this matrix, $R_B^F(\theta_C)$ represents a static mounting on the quadrotor and is given by

$$R_B^F = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}. \quad (8)$$

The camera positioner unit is considered to have coincident rotational links, thus the link lengths are zero ($a = 0$ in Table I).

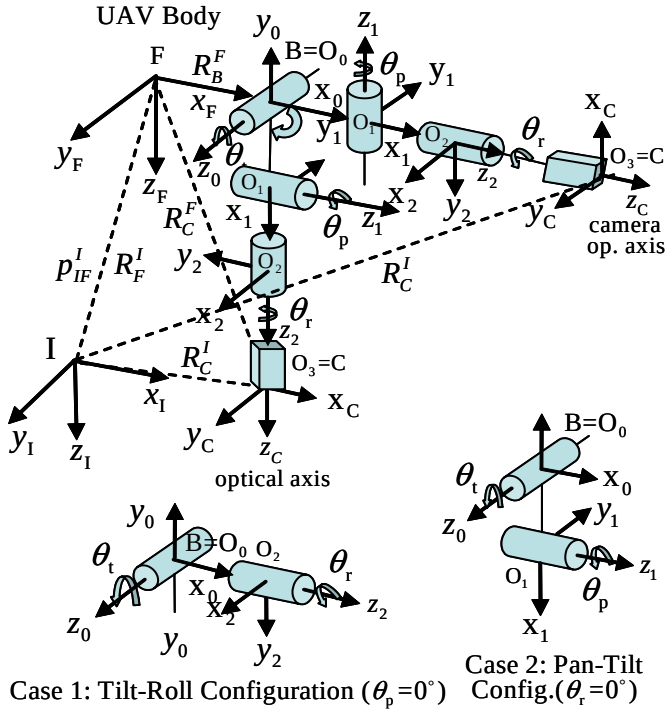


Fig. 3. Quadrotor with a Pan-Tilt-Roll camera positioner.

1) Case 1: Tilt-Roll Camera Configuration (camera looking forward): The Denavit-Hartenburg table for the Tilt-Roll configuration in Figure 3 is shown in Table I. The Tilt-Roll camera positioner is obtained by freezing the second manipulator joint. The link rotation matrices can be obtained using standard results [12] and setting $\theta_p(t) = 0$ yields the modified result

$$R_2^1 = \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} (\theta_p = 0^\circ). \tag{9}$$

The total rotation matrix from the quadrotor through the 3rd link can be obtained as

$$\begin{aligned} R_C^F &= R_B^F R_1^{B=O_0} R_2^1 R_{3=C}^2 \\ &= \begin{bmatrix} -\sin\theta_t \cos\theta_r & \sin\theta_t \sin\theta_r & \cos\theta_t \\ \sin\theta_r & \cos\theta_r & 0 \\ -\cos\theta_t \cos\theta_r & \cos\theta_t \sin\theta_r & -\sin\theta_t \end{bmatrix}. \end{aligned} \tag{10}$$

Link	d (offset)	a (length)	α_t (twist)	θ_a (angle)
1	0	0	-90°	θ_t
2	0	0	-90°	$\theta_p - 90^\circ$
3	0	0	0°	$\theta_r - 90^\circ$

TABLE I
Denavit-Hartenburg Table for a Tilt-Roll Camera Positioner

The angular velocity of the camera frame can be written as follows

$$\omega_{BC}^F = R_B^F \omega_{BC}^B = R_B^F J \dot{\theta}_C = J_C' \dot{\theta}_C^{'} \tag{11}$$

where the positioner joint angles, $\dot{\theta}_C^{'}(t) \in \mathbb{R}^3$, are given by

$$\dot{\theta}_C^{'} = \begin{bmatrix} \dot{\theta}_t \\ \dot{\theta}_p \\ \dot{\theta}_r \end{bmatrix} \text{ and } \theta_C^{'} = \begin{bmatrix} \theta_t \\ \theta_p \\ \theta_r \end{bmatrix} \text{ in which } \theta_p(t) = 0^\circ. \tag{12}$$

The Jacobian matrix $J_C'(\theta_C^{'}) \in \mathbb{R}^{3\times 3}$ in (11) can be built from the rotation matrices with the final result given by

$$J_C' = R_B^F \begin{bmatrix} z_0 & z_1 & z_2 \end{bmatrix} = \begin{bmatrix} 0 & -\sin\theta_t & \cos\theta_t \\ 1 & 0 & 0 \\ 0 & -\cos\theta_t & -\sin\theta_t \end{bmatrix} \tag{13}$$

where z_i are the columns of the rotation matrices as specified in [12]. By way of example, the third column in (8) is the z_0 vector in $J_C'(\theta_C^{'})$. To simplify the use of the Jacobian matrix in later developments, it can be noted that $\dot{\theta}_p(t) = 0$ (since it assumed to be constant) and hence a new reduced position joint angle vector can be introduced as

$$\dot{\theta}_C = \begin{bmatrix} \dot{\theta}_t \\ \dot{\theta}_r \end{bmatrix} \in \mathbb{R}^2 \text{ and } \theta_C = \begin{bmatrix} \theta_t \\ \theta_r \end{bmatrix} \in \mathbb{R}^2 \tag{14}$$

along with a reduced Jacobian matrix, $J_C(\theta_C) \in \mathbb{R}^{3\times 2}$, defined as

$$J_C = R_B^F \begin{bmatrix} z_0 & z_2 \end{bmatrix} = \begin{bmatrix} 0 & \cos\theta_t \\ 1 & 0 \\ 0 & -\sin\theta_t \end{bmatrix}. \tag{15}$$

The total rotation matrix from the camera frame to the inertial frame can be obtained using the previous equations as

$$R_C^I = R_F^I R_C^F \tag{16}$$

and the desired rotation matrix expressed in inertial frame, $R_{Cd}^I(\Theta, \theta_C^{'})$, can be defined in the following manner

$$R_{Cd}^I = R_F^I R_C^F R_{Cd}^C. \tag{17}$$

Hence, the rotation matrix between the camera frame and the desired frame can be obtained using measurable rotations as

$$\begin{aligned} R_{Cd}^C &= \left(R_C^I\right)^\intercal R_{Cd}^I \\ &= \left(R_F^C R_I^F\right) R_F^I R_C^F R_{Cd}^C. \end{aligned} \tag{18}$$

Remark 1: The more details are shown in Appendix B and a similar approach can be followed to develop the kinematics for the Pan-Tilt configuration.

C. Translation and Orientation of the Camera Frame

The objective is to control the motion of the camera optical axis. Towards this end, the kinematic relationships in (1) - (3) will be extended to include the action of the camera positioning unit and to obtain the position and orientation of the camera. The derivative of the position of the camera in the camera frame, C, with

respect to the inertial frame, I, and expressed in the inertial frame, $\dot{p}^I_{IC}(t) \in \mathbb{R}^3$, is defined as

$$\dot{p}^I_{IC} = R^I_C v^C_{IC} \tag{19}$$

where $v^C_{IC}(t) \in \mathbb{R}^3$ is the linear velocity in the camera frame, C, referred to inertial frame, I, and expressed in the camera frame, C. The velocity $v^C_{IC}(t)$ can be divided into two components as follows

$$\begin{aligned} R^I_C v^C_{IC} &= R^I_C \left(v^C_{IF} + v^C_{FC}\right) \\ &= R^I_C \left(R^C_F v^F_{IF} + R^C_F v^F_{FC}\right) \\ &= R^I_F v^F_{IF} \end{aligned} \tag{20}$$

where the fact that $v^F_{FC}(t) = 0$ is used, since the camera positioning unit only has rotational axes and does not translate from the quadrotor body. The desired camera position trajectory, $\dot{p}^I_{ICd}(t) \in \mathbb{R}^3$, is generated via

$$\dot{p}^I_{ICd} = R^I_C v^C_{ICd} \tag{21}$$

where $v^C_{ICd}(t) \in \mathbb{R}^3$ is a desired input velocity vector.

The second equation of (2) uses a Jacobian matrix $T^I_F(\Theta)$ to relate the rotational velocity in the quadrotor frame to the rotational velocity in the inertial frame. This relationship can be used to solve for $\Theta^I_{IF}(t)$ as

$$\Theta^I_{IF} = \int_0^t T^I_F(\Theta)\, \omega^F_{IF} dt \tag{22}$$

where $\Theta^I_{IF}(t) = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^\intercal \in \mathbb{R}^3$ represents the roll, pitch, and yaw angles between the quadrotor frame and inertial frame as described in [4]. The exact form of $T^I_F(\Theta)$ can be expressed as follows

$$T^I_F(\Theta) = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix}. \tag{23}$$

A similar result is now shown for the camera angle, $\Theta^I_{IC}(t)$. The Jacobian matrix $T^I_F(\Theta)$ is used to relate the rotational velocity of the camera in the quadrotor frame to the rotational velocity in the inertial frame as

$$\begin{aligned} \dot{\Theta}^I_{IC} &= T^I_F(\Theta)\, \omega^F_{IC} \\ &= T^I_F(\Theta) \left(R^F_C \omega^C_{IC}\right) \end{aligned} \tag{24}$$

and decomposing the angular velocity, $\omega^C_{IC}(t)$, yields

$$\begin{aligned} \dot{\Theta}^I_{IC} &= T^I_F(\Theta)\, R^F_C \left(\omega^C_{IF} + \omega^C_{FC}\right) \\ &= T^I_F(\Theta)\, R^F_C \left(\omega^C_{IF} + (\omega^C_{FB} + \omega^C_{BC})\right) \\ &= T^I_F(\Theta)\, R^F_C \left(R^C_F \omega^F_{IF} + R^C_F \omega^F_{FB} + R^C_F \omega^F_{BC}\right) \\ &= T^I_F(\Theta) \left(\omega^F_{IF} + R^F_B \omega^B_{BC}\right) \\ &= T^I_F(\Theta)\, \omega^F_{IF} + T^I_F(\Theta)\, R^F_B J \dot{\theta}_c \end{aligned} \tag{25}$$

where $R^F_C R^C_F = 1$ and $\omega^F_{FB}(t) = 0$ since the camera base is rigidly mounted on the quadrotor frame. The definition $J_C = R^F_B J(t)$ is introduced into (25) to yield

$$\dot{\Theta}^I_{IC} = T^I_F(\Theta)\, \omega^F_{IF} + T^I_F(\Theta)\, J_C \dot{\theta}_c \tag{26}$$

where $J_C \dot{\theta}_c(t)$ was defined in (15) as the camera kinematics. Finally, following the same approach the desired camera angle, $\Theta^I_{ICd}(t)$, is obtained from the desired angular velocity of the camera in the quadrotor frame, $\omega^C_{ICd}(t)$, as

$$\begin{aligned} \dot{\Theta}^I_{ICd} &= T^I_F(\Theta)\, \omega^F_{ICd} \\ &= T^I_F(\Theta)\, R^F_C \omega^C_{ICd}. \end{aligned} \tag{27}$$

The changing rate of $R^I_C(\Theta)$,i.e., $\dot{R}^I_C(\Theta)$, is obtained as follows

$$\dot{R}^I_C = R^I_C S(\omega^C_{IC}) \tag{28}$$

thus

$$\begin{aligned} \dot{R}^C_I &= \left[R^I_C S(\omega^C_{IC})\right]^\intercal \\ &= S^\intercal(\omega^C_{IC}) R^C_I \\ &= -S(\omega^C_{IC}) R^C_I. \end{aligned} \tag{29}$$

D. Model Assumptions

The following assumptions are made regarding the system model:

- A1: The position $p^I_{IF}(t)$ and velocities $v^F_{IF}(t)$, $\omega^F_{IF}(t)$ are measurable and the angles of the quadrotor and Pan-Tilt-Roll camera unit are also measurable.
- A2: The quadrotor frame expressed in F, camera base expressed in B, and camera frame C are all coincident since the link lengths and link offsets are assumed to have zero length..
- A3: $R^I_F(\Theta)$ and $T^I_F(\Theta)$ are full rank, i.e., $\theta(t) \neq \pm\frac{\pi}{2}$ so that [4]. This will ensure that the orientation angle, $\alpha(t)$, (defined later) remains within the range $0 \leq \alpha(t) < 2\pi$ about the rotation axis $\mu(t)$ and will ensure that $det(L_\omega)$ exists.
- A4: The desired camera trajectories and the first derivatives are all bounded; i.e., $p^I_{ICd}(t)$, $v^C_{ICd}(t)$, and $\omega^C_{ICd}(t) \in \mathcal{L}_\infty$.
- A5: $N_1(v^F_{IF})$ in (4) can be replaced by the linearly parameterized form $Y_1(v^F_{IF})\theta_1 = N_1(v^F_{IF})$ where $Y_1(v^F_{IF}) \in \mathbb{R}^{3\text{x}n}$ is a known regression matrix and $\theta_1 \in \mathbb{R}^n$ is a known parameter vector. Additionally, $\left\|Y_1(v^F_{IF})\theta_1\right\| \leq \zeta_1(\left\|v^F_{IF}\right\|) \leq \xi_1 \left\|v^F_{IF}\right\|$ where ζ_1 is a positive function and non-decreasing in $\left\|v^F_{IF}\right\|$ and $\xi_1 \in \mathbb{R}^1$ is a positive constant.

III. Control Method

Figure 4 demonstrates the culmination of the modeling effort to combine the quadrotor and camera positioner to create a means of fully actuating the camera optical axis. In this diagram, it can be seen that the camera is positioned and oriented by using the two camera positioner angles (selected from $\theta_{tilt}(t), \theta_{roll}(t)$, and $\theta_{pan}(t)$ according to configuration), the quadrotor linear force, $F(t)$, and the quadrotor angles $\phi_{roll}(t), \theta_{pitch}(t)$, and $\psi_{yaw}(t)$ by $\omega_x(t), \omega_y(t)$, and $\omega_z(t)$ from (2). In keeping with the fly-the-camera objective, a controller will be designed based on these inputs to move the camera optical axis along a desired trajectory. A control strategy

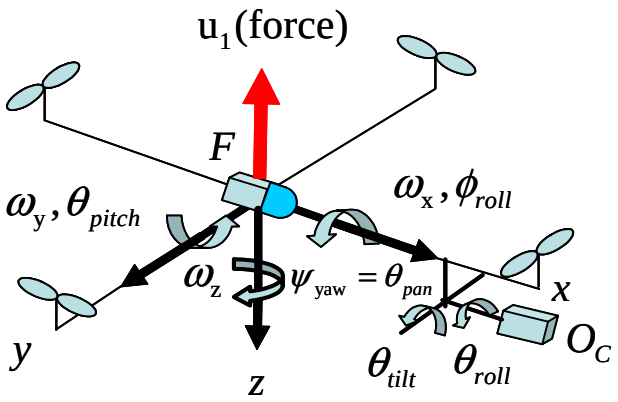


Fig. 4. The Camera Optical Axis is fully actuated by two camera positioner angles, a linear force, and the three angular velocities.

will be proposed to control the camera translational position error, $e_p(t) \in \mathbb{R}^3$, and the camera orientation error, $e_\theta(t) \in \mathbb{R}^3$.

A. Open-Loop Error Formulation

The camera translational position error, $e_p(t)$, is defined in the camera frame (C) as the transformed difference between the inertial frame (I) based camera position, $p^I_{IC}(t)$, and the inertial frame based desired camera position, denoted as $p^I_{ICd}(t) \in \mathbb{R}^3$, as follows

$$e_p \triangleq R^C_I(p^I_{IC} - p^I_{ICd}). \tag{30}$$

The camera translational position error rate, $\dot{e}_p\ (t) \in \mathbb{R}^3$, is obtained by taking the time derivative of (30) to yield

$$\dot{e}_p = \dot{R}^C_I(p^I_{IC} - p^I_{ICd}) + R^C_I(\dot{p}^I_{IC} - \dot{p}^I_{ICd}) \tag{31}$$

where $\dot{p}^I_{IC}(t)$ and $\dot{p}^I_{ICd}(t)$ were introduced in (19) and (21), respectively. Substituting (29) into the first term in (31) yields

$$\begin{aligned} \dot{R}^C_I(p^I_{IC} - p^I_{ICd}) &= -S(\omega^C_{IC})R^C_I(p^I_{IC} - p^I_{ICd}) \\ &= -S(\omega^C_{IC})e_p. \end{aligned} \tag{32}$$

and the term $R^C_I\dot{p}^I_{IC}(t)$ in (31) can be rewritten in terms of the quadrotor velocity, $v^F_{IF}(t)$, as

$$\begin{aligned} R^C_I\dot{p}^I_{IC} &= R^C_I(R^I_C v^C_{IC}) \\ &= v^C_{IC} \\ &= v^C_{IF} + v^C_{FC} \\ &= R^C_F v^F_{IF} \end{aligned} \tag{33}$$

with $v^C_{FC} = 0$, i.e., all translation of the camera is the result of the quadrotor translation. The final term in (31) is rewritten as

$$\begin{aligned} R^C_I\dot{p}^I_{ICd} &= R^C_I(R^I_C v^C_{ICd}) \\ &= v^C_{ICd}. \end{aligned} \tag{34}$$

Substituting (32), (33), and (34) into (31) yields

$$\dot{e}_p = -S(\omega^C_{IC})e_p + R^C_F v^F_{IF} - v^C_{ICd}. \tag{35}$$

To further the controller development, a filtered error, $r(t) \in \mathbb{R}^3$, is introduced as

$$r = \begin{bmatrix} r_p \\ e_\theta \end{bmatrix} \tag{36}$$

where the filtered position error, $r_p(t) \in \mathbb{R}^3$, is defined as

$$r_p = k_p e_p + R^C_F v^F_{IF} + R^C_F \delta \tag{37}$$

in which $\delta = [\ 0 \quad 0 \quad \delta_3\ ]^\intercal \in \mathbb{R}^3$ is a constant design vector. The orientation tracking error signal, $e_\theta(t) \in \mathbb{R}^3$, is defined in terms of the angle-axis representation of the rotation matrix between the desired and actual camera orientations, $R^C_{Cd}(t)$, [1] as

$$e_\theta = \alpha\mu \tag{38}$$

where the scalar angle $\alpha(\Theta)$ is obtained from

$$\alpha = \cos^{-1}\left(\frac{1}{2}(Tr(R^C_{Cd}) - 1)\right) \in \mathbb{R}^1, \tag{39}$$

in which $Tr(R^C_{Cd})$ defines the trace of the matrix $R^C_{Cd}(\Theta)$, and the unit length axis of rotation $\mu(\Theta) \in \mathbb{R}^3$ defined as

$$\mu = \frac{1}{2\sin\alpha}\begin{bmatrix} (r_{32} - r_{23}) \\ (r_{13} - r_{31}) \\ (r_{21} - r_{12}) \end{bmatrix} \tag{40}$$

for which $\|\mu\|^2 = 1$. In this representation, the rotation angle $\alpha(\Theta)$ is assumed to stay within the range $0 \leq \alpha(\Theta) < 2\pi$. Note that the terms on the right-hand side of the definition in (40) come from $R^C_{Cd}(\Theta)$ as

$$R^C_{Cd} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}. \tag{41}$$

Substituting the axis-angle representation from (39) and (40) into (38) yields

$$e_\theta = \frac{1}{2\ sinc\{\cos^{-1}(\frac{1}{2}(Tr(R^C_{Cd}) - 1))\}}\begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix} \tag{42}$$

where $sinc(\alpha) = \dfrac{\sin(\alpha)}{\alpha}$.

The angular error rate can be obtained by taking the time derivative of (38) as

$$\dot{e}_\theta = L_\omega \omega^C_{CCd} \in \mathbb{R}^3 \tag{43}$$

where

$$L_\omega = I_3 - \frac{\alpha}{2}S(\mu) + \left(1 - \frac{sinc(\alpha)}{sinc^2(\frac{\alpha}{2})}\right)S^2(\mu) \tag{44}$$

in which $S(\mu)$ is a skew symmetric matrix defined as

$$S(\mu) = \frac{(R^C_{Cd} - (R^C_{Cd})^\intercal)}{2\sin\alpha}. \tag{45}$$

Details for obtaining of (43) - (45) can be found in [8] and [5]. The term $\omega^C_{CCd}(t) \in \mathbb{R}^3$ introduced in (43) represents

the angular velocity of the desired camera frame relative to the actual camera frame as

$$\begin{aligned}\omega^C_{CCd} &= \omega^C_{CI} + \omega^C_{ICd} = -\omega^C_{IC} + \omega^C_{ICd} = -R^C_F\omega^F_{IC} + \omega^C_{ICd} \\ &= -R^C_F(\omega^F_{IF} + \omega^F_{FB} + \omega^F_{BC}) + \omega^C_{ICd} \\ &= \omega^C_{ICd} - R^C_F(\omega^F_{IF} + \omega^F_{BC}).\end{aligned} \tag{46}$$

where the rigid connection of the manipulator to the base requires $\omega^F_{FB}(t) = 0$. Substituting (46) along with (11), (14), and (15) into (43) produces

$$\begin{aligned}\dot{e}_\theta &= L_\omega\left(\omega^C_{ICd} - R^C_F\omega^F_{IF} - R^C_F\omega^F_{BC})\right) \\ &= L_\omega\omega^C_{ICd} - L_\omega R^C_F\omega^F_{IF} - L_\omega R^C_F J_c\dot{\theta}_c.\end{aligned} \tag{47}$$

Taking the time derivative of $r_p(t)$ in (37) yields

$$\dot{r}_p = R^C_F\dot{v}^F_{IF} + \dot{R}^C_F v^F_{IF} + k_p\dot{e}_p + \dot{R}^C_F\delta. \tag{48}$$

Substituting from (4) and (35) into (48), utilizing the fact that $\dot{R}^C_F(\Theta) = R^C_F S(\omega^F_{CF}) = -R^C_F S(\omega^F_{FC})$, and subtracting and adding $R^C_F S(\omega^F_{IF})\delta$ yields

$$\begin{aligned}\dot{r}_p &= R^C_F\left[\tfrac{1}{m}N_1(v^F_{IF}) - S(\omega^F_{IF})v^F_{IF} + gR^F_I E_3 + \tfrac{1}{m}F^F_f\right] \\ &\quad + k_p\left[R^C_F v^F_{IF} - S(\omega^C_{IC})e_p - v^C_{ICd}\right] + R^C_F S(\omega^F_{IF})\delta \\ &\quad - R^C_F S(\omega^F_{FC})v^F_{IF} - R^C_F\left(S(\omega^F_{FC}) + S(\omega^F_{IF})\right)\delta\end{aligned} \tag{49}$$

Combining the angular velocities represented by the second term in the first row of (49) and the first term in the last row of (49) yields

$$\begin{aligned}&-R^C_F\left(S(\omega^F_{IF}) + S(\omega^F_{FC})\right)v^F_{IF} \\ &= -R^C_F S\left(\omega^F_{IF} + \omega^F_{FC}\right)v^F_{IF} \\ &= -R^C_F S(\omega^F_{IC})v^F_{IF}.\end{aligned} \tag{50}$$

The right-hand side of (50) can be further clarified using $\omega^F_{IC} = R^F_C\omega^C_{IC}(t)$ and

$$\begin{aligned}S(\omega^F_{IC}) &= S(R^F_C\omega^C_{IC}) \\ &= R^F_C S(\omega^C_{IC})R^C_F.\end{aligned} \tag{51}$$

to yield

$$\begin{aligned}&-R^C_F\left(S(\omega^F_{IF}) + S(\omega^F_{FC})\right)v^F_{IF} \\ &= -R^C_F R^F_C S(\omega^C_{IC})R^C_F v^F_{IF} \\ &= -S(\omega^C_{IC})R^C_F v^F_{IF}.\end{aligned} \tag{52}$$

Combining the angular velocity terms $\omega^F_{FC}(t)$ and $\omega^F_{IF}(t)$ in the last terms of the last row in (49) with $\omega^F_{IC}(t)$ yields

$$\begin{aligned}-R^C_F\left(S(\omega^F_{FC}) + S(\omega^F_{IF})\right)\delta &= -R^C_F S\left(\omega^F_{IF} + \omega^F_{FC}\right)\delta \\ &= -R^C_F S(\omega^F_{IC})\delta \\ &= -R^C_F S(R^F_C\omega^C_{IC})\delta \\ &= -R^C_F R^F_C S(\omega^C_{IC})R^C_F\delta \\ &= -S(\omega^C_{IC})R^C_F\delta.\end{aligned} \tag{53}$$

Multiplying both sides of (37) by $S(\omega^C_{IC})$ yields

$$-S(\omega^C_{IC})r_p = -S(\omega^C_{IC})\left(k_p e_p + R^C_F v^F_{IF} + R^C_F\delta\right). \tag{54}$$

Substituting (52), (53), and (54) into (49) yields

$$\begin{aligned}\dot{r}_p &= \tfrac{1}{m}R^C_F N_1(v^F_{IF}) - S(\omega^C_{IC})r_p + gR^C_I E_3 + \tfrac{1}{m}R^C_F F^F_f \\ &\quad + k_p R^C_F v^F_{IF} - k_p v^C_{ICd} - R^C_F S(\delta)\omega^F_{IF}.\end{aligned} \tag{55}$$

By taking the time derivative of $r(t)$ in (36) and substitute (47) and (55) it can be obtained that

$$\begin{aligned}\dot{r} &= \begin{bmatrix}\dot{r}_p \\ \dot{e}_\theta\end{bmatrix} \\ &= \begin{bmatrix}-S(\omega^C_{IC})r_p \\ O_{3x1}\end{bmatrix} + \begin{bmatrix}\frac{1}{m}R^C_F F^F_f - R^C_F S(\delta)\omega^F_{IF} \\ -L_\omega R^C_F\omega^F_{IC} - L_\omega R^C_F J_c\dot{\theta}_c\end{bmatrix} \\ &+ \begin{bmatrix}\frac{1}{m}R^C_F N_1(v^F_{IF}) + k_p R^C_F v^F_{IF} + gR^C_I E_3 \\ O_{3x1}\end{bmatrix} \\ &- \begin{bmatrix}k_p v^C_{ICd} \\ L_\omega\omega^C_{ICd}\end{bmatrix}.\end{aligned} \tag{56}$$

Arranging the last term in the first row of (56) yields

$$\begin{aligned}&\begin{bmatrix}\frac{1}{m}R^C_F F^F_f - R^C_F S(\delta)\omega^F_{IF} \\ -L_\omega R^C_F\omega^F_{IC} - L_\omega R^C_F J_c\dot{\theta}_c\end{bmatrix} \\ &= \begin{bmatrix}R^C_F & O_{3x3} \\ O_{3x3} & -L_\omega R^C_F\end{bmatrix}\begin{bmatrix}\frac{1}{m}B_1, -S(\delta), O_{3x2} \\ O_{3x1},\ I_{3x3},\ J_c\end{bmatrix}\begin{bmatrix}u_1 \\ \omega^F_{IF} \\ \dot{\theta}_c\end{bmatrix} \\ &= \bar{L}_\omega B\bar{U}\end{aligned} \tag{57}$$

where $\bar{L}_\omega \in \mathbb{R}^{6x6}$ is defined as

$$\bar{L}_\omega = \begin{bmatrix}R^C_F & O_{3x3} \\ O_{3x3} & -L_\omega R^C_F\end{bmatrix}, \tag{58}$$

$B(t) \in \mathbb{R}^{6x6}$ is defined as

$$B = \begin{bmatrix}\frac{1}{m}B_1 & -S(\delta) & O_{3x2} \\ O_{3x1} & I_{3x3} & J_c\end{bmatrix} \tag{59}$$

$$= \begin{bmatrix}0 & 0 & +\delta_3 & -\delta_2 & 0 & 0 \\ 0 & -\delta_3 & 0 & +\delta_1 & 0 & 0 \\ \frac{1}{m} & +\delta_2 & -\delta_1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \cos\theta_t \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & -\sin\theta_t\end{bmatrix}, \tag{60}$$

and $\bar{U}(t) \in \mathbb{R}^6$ is given by

$$\bar{U} = \begin{bmatrix}u_1 \\ \omega^F_{IF} \\ \dot{\theta}_c\end{bmatrix}. \tag{61}$$

It should be noted that the final form of $B(t)$ in (59) represents the point where the specifics of the Tilt-Roll Jacobian, $J_c(t)$, are explicitly used, the Pan-Tilt configuration will require a different $B(t)$. The new term $\bar{B}(t)$ is introduced as

$$\bar{B} = \bar{L}_\omega B \tag{62}$$

along with the new term $U(t) = \begin{bmatrix}U_1^\intercal & U_2^\intercal\end{bmatrix}^\intercal \in \mathbb{R}^6$, where $U_1(t) \in \mathbb{R}^3$ and $U_2(t) \in \mathbb{R}^3$, defined also further clarify the control design procedure.

$$U = \bar{B}\bar{U}. \tag{63}$$

Substitution of (62) and (63) into (57) and then substitution of the resulting form of (57) into (56) produces the open-loop filtered error dynamics as follows

$$\begin{aligned}\dot{r} &= \begin{bmatrix}-S(\omega^C_{IC})r_p \\ O_{3x1}\end{bmatrix} + \begin{bmatrix}U_1 \\ U_2\end{bmatrix} - \begin{bmatrix}k_p v^C_{ICd} \\ L_\omega\omega^C_{ICd}\end{bmatrix} \\ &+ \begin{bmatrix}\frac{1}{m}R^C_F N_1(v^F_{IF}) + k_p R^C_F v^F_{IF} + gR^C_I E_3 \\ O_{3x1}\end{bmatrix}\end{aligned} \tag{64}$$

where $R_F^C R_I^F = I_3$ is used and it is clear that the control inputs $U_1(t)$ and $U_2(t)$ will be designed to meet the control objective. Note that implementation of the control while require $\bar{U}(t)$ which is obtained from

$$\bar{U} = \bar{B}^{-1}U \in \mathbb{R}^6 \tag{65}$$

requires $\bar{B}^{-1} = B^{-1}(\bar{L}_\omega)^{-1}$ where

$$(\bar{L}_\omega)^{-1} = \frac{1}{\Delta(\bar{L}_\omega)} \begin{bmatrix} -L_\omega R_F^C & O_{3x3} \\ O_{3x3} & R_F^C \end{bmatrix} \tag{66}$$

in which $\Delta(\bar{L}_\omega) = -R_F^C L_\omega R_F^C$ is the determinant of the matrix $\bar{L}_\omega(t)$ and $\Delta(\bar{L}_\omega) \neq O_{6x6}$ due to Assumption A3.

B. Control Design

The non-negative scalar function $V(t)$ is chosen as

$$V = \frac{1}{2} r^\intercal r + \frac{1}{2} e_p^\intercal e_p. \tag{67}$$

Differentiating yields

$$\dot{V} = r^\intercal \dot{r} + e_p^\intercal \dot{e}_p, \tag{68}$$

by substituting (35) and (64) into (68) it can be obtained that

$$\begin{array}{l} \dot{V} = [r_p^\intercal, e_\theta^\intercal] \left\{ \begin{bmatrix} -S(\omega_{IC}^C) r_p \\ O_{3x1} \end{bmatrix} + \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} + \right. \\ \left. \begin{bmatrix} \frac{1}{m} R_F^C N_1(v_{IF}^F) + k_p R_F^C v_{IF}^F + g R_I^C E_3 - k_p v_{ICd}^C \\ L_\omega \omega_{ICd}^C \end{bmatrix} \right\} \\ + e_p^\intercal \left[-S(\omega_{IC}^C) e_p + \left(r_p - k_p e_p - R_F^C \delta \right) - v_{ICd}^C \right] \end{array} \tag{69}$$

where (37) was utilized. The terms in (69) can be collected to yield

$$\begin{array}{l} \dot{V} = [-r_p^\intercal S(\omega_{IC}^C) r_p + \frac{1}{m} r_p^\intercal R_F^C N_1(v_{IF}^F) + r_p^\intercal k_p R_F^C v_{IF}^F \\ + r_p^\intercal U_1 + r_p^\intercal g R_I^C E_3 - r_p^\intercal k_p v_{ICd}^C + e_p^\intercal r_p - e_p^\intercal S(\omega_{IC}^C) e_p \\ - k_p e_p^\intercal e_p - e_p^\intercal R_F^C \delta - e_p^\intercal v_{ICd}^C;\ e_\theta^\intercal U_2 - e_\theta^\intercal L_\omega \omega_{ICd}^C]. \end{array} \tag{70}$$

Equation (70) will be utilized to design the control inputs $U_1(t)$ and $U_2(t)$. From the upper equation in (70) we can design $U_1(t)$ to subtract out four terms, add stabilizing feedback, and add robust compensatation for the unknown nonlinear term as follows

$$U_1 = -k_r r_p - \frac{r_p \zeta_1^2(\|v_{IF}^F\|)}{\varepsilon_0} + k_p R_F^C v_{IF}^F - g R_I^C E_3 - e_p \tag{71}$$

where ε_0 is a positve constant, Assumption 5 is used for $\zeta_1\left(\|v_{IF}^F\|\right)$, and the nonlinear term $\bar{N}_1(t)$ is defined by

$$\bar{N}_1 = \frac{1}{m} R_F^C N_1(v_{IF}^F). \tag{72}$$

From the lower equation in (64) $U_2(t)$ can be designed as

$$U_2 = L_\omega \omega_{ICd}^C - k_\theta e_\theta. \tag{73}$$

Next, substituting these designed control inputs (71) and (73) into (64), we can get the closed-loop filterd error design based on Lyapunov stability analysis as follows

$$\dot{r} = [-S(\omega_{IC}^C) r_p - k_r r_p - \frac{r_p \zeta_1^2(\|v_{IF}^F\|)}{\varepsilon_1} + \bar{N}_1 - k_p v_{ICd}^C;\ -k_\theta e_\theta]. \tag{74}$$

Substituting (71) and (73) into (70) yields

$$\begin{array}{l} \dot{V} = r_p^\intercal (\bar{N}_1 - \frac{r_p \zeta_1^2(\|v_{IF}^F\|)}{\varepsilon_0}) - k_r r_p^\intercal r_p - r_p^\intercal k_p v_{ICd}^C - k_\theta e_\theta^\intercal e_\theta \\ - r_p^\intercal S(\omega_{IC}^C) r_p - e_p^\intercal S(\omega_{IC}^C) e_p - k_p e_p^\intercal e_p - e_p^\intercal (R_F^C \delta + v_{ICd}^C). \end{array} \tag{75}$$

IV. Stability Analysis

Theorem 1: The closed-loop control law of (71) and (73) ensure that the tracking error is Globally Uniformly Ultimately Bounded (GUUB) in the manner

$$\|\eta\| \leq \sqrt{\|\eta(0)\|^2 e^{-2\lambda_2 t} + \frac{\varepsilon_4}{2\lambda_2}} \tag{76}$$

where

$$\eta \triangleq [r_p^T, e_\theta^T, e_p^T]^T, \tag{77}$$

ε_4 is a positive constant, and λ_2 is a positive constant given by the following form

$$\lambda_2 = \min\{(k_r - \tfrac{\lambda_1}{2}), \left(k_p - \tfrac{\lambda_0}{2}\right), k_\theta\} \tag{78}$$

where λ_0, λ_1 are positive constants under the conditions that

$$k_r > \tfrac{\lambda_1}{2} \text{ and } k_{p>} \tfrac{\lambda_0}{2}. \tag{79}$$

A proof of the theorem is given in Appendix A.

V. Simulation

A two computer system was built to simulate the proposed controller as shown in Figure 5. The first computer is configured to run QNX Real-Time Operating System (RTOS) and host the QMotor [11] control and simulation package while the second computer is configured to run Windows XP and host the FlightGear (v0.9.10) [3] open-source flight simulator package. A QMotor program was written to simulate the rigid body kinematics dynamics and the feedback control. The output of the dynamics simulation is sent via UDP to set the aircraft/camera position and orientation in the FlightGear virtual world (note that FlightGear is used only as a graphics processor). The desired trajectories are input by the operator using a 6DOF joystick (Logitech Extreme 3D Pro [6]). Specifically, the operator indirectly supplies the desired position trajectory used by the controller through monitoring the simulated camera view and using the joystick to command the velocities that move the camera view in the virtual world. The three inputs on the joystick, labeled as x, y, and twist, are used to generate and control either three translational velocities, $v_{ICd}^C(t)$, or the three angular velocities, $\omega_{ICd}^C(t)$ depending on trigger position. That is, the magnitude and direction of these quantities is derived from the joystick position. The velocity commands are then integrated to produce the desired position trajectory used by the controller. A typical scene from FlightGear is shown in Figure 6 where the quadrotor is tilted but the camera view seen by the operator remains level.

The quadrotor simulation was developed to approximate the parameters of the DraganFlyer X-Pro [2]. Parameters such as mass (m) and saturation limits for

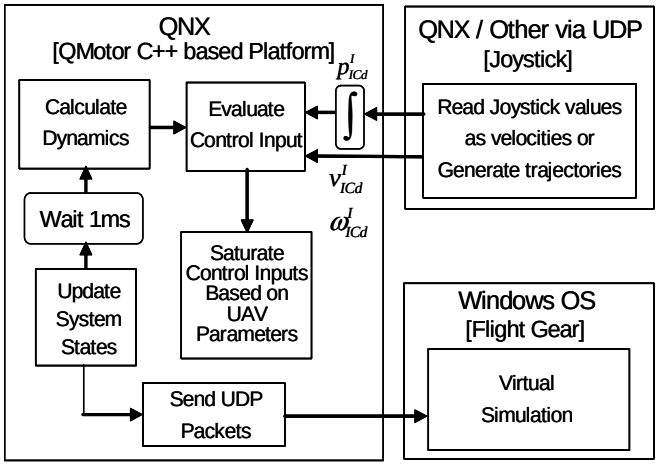


Fig. 5. Overview of components in simulation system.

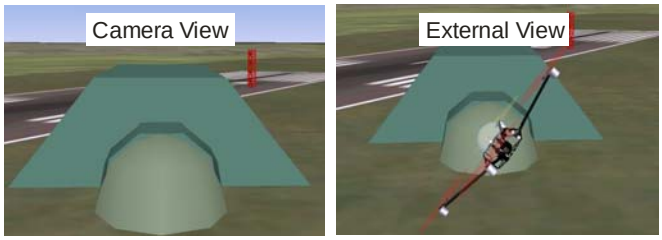


Fig. 6. The "fly-the-camera" view used by the operator and an outside view of the quadrotor position.

control inputs are found in [10]. The control gains are chosen to be

$$\begin{aligned} k_r &= k_\theta = diag(1,1,1),\ k_p = diag(1,1,5), \\ m &= 2.72[kg] \text{ and } g = 9.81[m^3/kgs^2], \\ u_{1\_\max} &= 35.586\ [Nm], \\ \theta_{tilt\_\max} &= \theta_{roll\_\max} = 4.067\ [Nm] \end{aligned} \tag{80}$$

In order to orient the simulated quadrotor and camera systems with the virtual world, a desired rotation matrix between the camera frame (C) and Flight Gear frame (G) was defined as

$$R^C_{Cd=G} = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}. \tag{81}$$

A short timespan of the simulation was captured to demonstrate the operation of the system. The simulation results shown in Figure 15 and Figure 16 and displayed in Figure 1 demonstrate that the camera and quadrotor move in opposite directions to acheive the fly-the-camera perspective. Figure 7 shows the position errors about the coordinates (x, y, z) and Figure 8 shows the position tracking of the quad-rotor to the desired trajectory $p_d(t)$ in (30). The actual quad-rotor trajectory represented by the blue line follows the desired trajectory represented by the red line which is commanded to go up at the first time and then, move to forward and again go forward near the end to the $p_d(t)$. Figure 9 shows the filtered tracking errors in (37). Figure 10 shows the control inputs. The translational force input $u_1(t)$ of the quadrotor is collectively steady after having transient states in order to track the desired trajectory. Torque inputs are given to turn left, right, and to tilt. The camera torque inputs in the two bottom figures are shown for tilting and rolling. Figure 11 shows the velocity tracking errors in (35) by the results of Figure 12. Figure 13 shows the angular velocity tracking. Figure 14 shows the angle-axis signal of (38) in camera frame. Figure 15 shows the angles of the Tilt-Roll manipulator in (14) and Figure 16 shows the Euler angles about roll, pitch, and yaw of quadrotor quadrotor; $\Theta^I_{IF}(t)$ in (23).

VI. Conclusion

This paper suggests a novel fly-the-camera approach to designing a nonlinear controller for an underactuated quadrotor aerial vehicle that compliments the quadrotor motion with two additional camera axes to produce a fully actuated camera targeting platform. The fly-the-camera approach should provide a more intuitive perspective for a remote pilot to operate the quadrotor vehicle and camera for surveillance and navigation tasks. The approach fuses the often separate tasks of vehicle navigation and camera targeting into a single task where the pilot sees (and flies) the system as through riding on the camera optical axis. The controller was shown to provide position and angle tracking in the form of Globally Uniform Ultimately Bounded (GUUB) result using measurable position and velocity information as state feedback. Simulation results were shown as initial validation of the proposed system which are shown in Appendix C.

References

[1] V. Chitrakaran, D. Dawson, and H. Kannan. "Vision Assisted Autonomous Path Following for Unmanned Aerial Vehicles", Proc. of 45th IEEE Conference on Decision and Control, Dec. 13-15, 2006, pp. 63 - 68.
[2] DraganFlyer X-Pro: http://www.rctoys.com/
[3] "FlightGear", http://www.flightgear.org, cited 2/1/2007.
[4] T. I. Fossen, Marine Control Systems : Guidance, Navigation, and Control of Ships, Rigs, and Underwater Vehicles, Marine Cybernetics, 2003.
[5] DongBin Lee, Vilas Chitrakaran, Timorthy C. Burg, Darren M. Dawson, and Bin Xian "Control of a Remotely Operated Quadrotor Aerial Vehicle and Camera Unit Using a Fly-The-Camera Perspective", CRB Technical Report, CU/CRB/3/11/07/#1, http://www.ece.clemson.edu/crb/publictn/tr.htm, 2007.
[6] Logitech Extreme 3D Pro Joystick. http://www.logitech.com
[7] E. Malis and F. Chaumette, "2 1/2 D Visual Servoing with Respect to Unknown Objects Through a New Estimation Scheme of Camera Displacement", International Journal of Computer Vision, 37(1):79-97, June 2000.
[8] E. Malis, "Controbutions a la modelisation et a la commande en asservissement visuel," Ph.D. Thesis, University of Rennes I, IRISA, France, 1998.
[9] A. Mokhtari and A. Benallegue, "Dynamic Feedback Controller of Euler Angles and Wind Parameters Estimation for a quad-rotor Unmanned Aerial Vehicle", Proc. 2004 IEEE International Conference on Robotics and Automation, Vol 3, May 2004, pp. 2359 - 2366.

[10] A. Neff, D. Lee, V. Chitrakaran, D. Dawson, T. Burg, "Velocity Control of a Quad-Rotor UAV Fly-By-Camera Interface", Accepted, Proceedings of IEEE SoutheastCon 2007, Mar. 2007, to appear..
[11] "QMotor Real-Time Control Environment", http://www.ece.clemson.edu/crb/research/realtimesoftware/qmotor/index.htm, cited 2/1/2007.
[12] M. W. Spong, S. Hutchinson, and M. Vidyasagar, Robot Modeling and Control, John Wiley and Sons, Inc:, 2006.
[13] R. Stanciu and P. Oh, "Human-in-the-Loop Camera Control for a Mechatronic Broadcast Boom", IEEE/ASME Transactions on Mechatronics, Vol. 12, No. 1, February 2007.
[14] N. Cooke, H. Pringle, H. Pedersen, O. Connor, Human Factors of Remotely Operated Vehicles, ELSVIER JAI, 2006

Appendix A
Proof of Theorem 1

An upper bound for $\dot{V}(t)$ in (75) can be formed by first noting that the skew symmetry of $S(\cdot)$ implies that $r_p^{\intercal} S(\omega_{IC}^C) r_p(t) = 0$ and $e_p^{\intercal} S(\omega_{IC}^C) e_p(t) = 0$, and then specifying upper bounds as

$$\varepsilon_1 \geq \left\| -R_F^C \delta - v_{ICd}^C \right\| \tag{82}$$

and

$$\varepsilon_2 \geq \left\| k_p v_{ICd}^C \right\| \tag{83}$$

where ε_1 and ε_2 are positive scalar constants. An upper bound for $\dot{V}(t)$ can now be written as

$$\begin{aligned} \dot{V} \quad \leq \quad & -k_r \left\| r_p \right\|^2 - k_\theta \left\| e_\theta \right\|^2 - k_p \left\| e_p \right\|^2 + \left\| e_p \right\| \varepsilon_1 \\ & + \left\| r_p \right\| \varepsilon_2 + \left\| r_p \right\| \left\| \zeta_1(\cdot) \right\| \left(1 - \left\| r_p \right\| \frac{\left\| \zeta_1(\cdot) \right\|}{\varepsilon_0} \right) \end{aligned} \tag{84}$$

Bounds on individual terms in (84), specifically,

$$\left\| e_p \right\| \varepsilon_1 \leq \frac{1}{2} \left(\lambda_0 \left\| e_p \right\|^2 + \tfrac{1}{\lambda_0} \varepsilon_1^2 \right) \tag{85}$$

where λ_0 is a positive scalar constant,

$$\left\| r_p \right\| \varepsilon_2 \leq \frac{1}{2} \left(\lambda_1 \left\| r_v \right\|^2 + \tfrac{1}{\lambda_1} \varepsilon_2^2 \right) \tag{86}$$

where λ_1 is a positive scalar constant, and

$$\varepsilon_3 \geq \left\| r_p \right\| \left\| \zeta_1(\cdot) \right\| \left(1 - \left\| r_p \right\| \frac{\left\| \zeta_1(\cdot) \right\|}{\varepsilon_0} \right) \tag{87}$$

can be used to write a new upper bound for $\dot{V}(t)$ as

$$\begin{aligned} \dot{V} \quad \leq \quad & -(k_r - \tfrac{\lambda_1}{2}) \left\| r_p \right\|^2 - k_\theta \left\| e_\theta \right\|^2 \\ & -(k_p - \tfrac{\lambda_0}{2}) \left\| e_p \right\|^2 + \tfrac{\varepsilon_1^2}{2\lambda_0} + \tfrac{\varepsilon_2^2}{2\lambda_1} + \varepsilon_3. \end{aligned} \tag{88}$$

If a positive scalar constant λ_2 is selected according to (78), then we can obtain

$$\dot{V} \leq -\lambda_2 \left[\left\| r_p \right\|^2 + \left\| e_\theta \right\|^2 + \left\| e_p \right\|^2 \right] + \tfrac{\varepsilon_1^2}{2\lambda_0} + \tfrac{\varepsilon_2^2}{2\lambda_1} + \varepsilon_3, \tag{89}$$

and then the definition of $\eta(t)$ in (77) is used to form a final upper bound on $\dot{V}(t)$ as

$$\dot{V} \leq -\lambda_2 \left\| \eta \right\|^2 + \tfrac{\varepsilon_1^2}{2\lambda_0} + \tfrac{\varepsilon_2^2}{2\lambda_1} + \varepsilon_3. \tag{90}$$

The definition of $V(t)$ in (67) can be substituted into (90) to yield

$$\dot{V} \leq -2\lambda_2 V + \varepsilon_4 \tag{91}$$

where ε_4 is a new bounding constant introduced as

$$\tfrac{\varepsilon_1^2}{2\lambda_0} + \tfrac{\varepsilon_2^2}{2\lambda_1} + \varepsilon_3 \leq \varepsilon_4. \tag{92}$$

The solution to this differential inequality is

$$V = V(0) e^{-2\lambda_2 t} + \tfrac{\varepsilon_4}{2\lambda_2} \tag{93}$$

and the initial value of $V(0)$ is upper bounded as

$$V(0) \leq \tfrac{1}{2} \left\| \eta(0) \right\|^2. \tag{94}$$

Hence, $V(t)$ can be written as

$$V(t) \leq \tfrac{1}{2} \left\| \eta(0) \right\|^2 e^{-2\lambda_2 t} + \tfrac{\varepsilon_4}{2\lambda_2}. \tag{95}$$

Relying again on the definition of $V(t)$ in (67), a bound on $\eta(t)$ can be written as

$$\left\| \eta \right\| \leq \sqrt{2V(t)}, \tag{96}$$

and then (93) substituted to yield

$$\left\| \eta \right\| \leq \sqrt{\left\| \eta(0) \right\|^2 e^{-2\lambda_2 t} + \tfrac{\varepsilon_4}{2\lambda_2}} \tag{97}$$

which proves the result in Theorem 1.

Remark 2: According to Theorem 1 and its subsequent stability analysis $V(\eta(t))$ is bounded provided the controller gains are selected to satisfy (79). Based on the definition of in $\eta(t)$ in (77) it is possible to conclude that $e_p(t)$, $e_\theta(t)$, and $r_p(t)$ are bounded. The desired trajectories, $p_{ICd}^I(t)$, $v_{ICd}^C(t)$, and $\omega_{ICd}^C(t)$ are bounded by design. It was also assumed that the quadrotor pitch angle, $\theta(t) \neq \pm\frac{\pi}{2}$, so that $T_F^I(\Theta)$ is invertible and bounded from (23) and $R_F^I(\Theta)$ is bounded and full rank according to Assumption 3. The rotation matrix $R_C^F(\theta_c)$ is of full rank and bounded as shown in (10), and also $J_C^{'}(t)$ and $J_C(t)$ are full rank from (13) and (15). The product of $R_F^I(\Theta)$ and $R_C^F(\theta_c)$, $R_C^I(\Theta, \theta_c)$, is bounded and full rank. Thus, $p_{IC}^I(t)$ in (30), $\dot{p}_{ICd}^I(t)$ in (21), $G(R_F^I)$ in (5), and $v_{IF}^F(t)$ in (37) are bounded. Then $v_{IC}^C(t)$ in (20), $\dot{p}_{IC}^I(t)$ in (19), and $\dot{p}_{IF}^I(t)$ in (1) are bounded. Owing to the desired trajectory $\omega_{ICd}^I(t) \in \mathcal{L}_\infty$, $\dot{\Theta}_{ICd}^I$ is bounded from (27). Since $e_\theta(t)$ is bounded, $\mu(t) \in \mathcal{L}_\infty$ under Assumption 3 resulting in $L_\omega(t) \in \mathcal{L}_\infty$ in (44). Since $v_{IF}^F(t) \in \mathcal{L}_\infty$, the nonlinearity of the aerodynamic damping term, $N_1(v_{IF}^F)$ in (4) is upper bounded by $\zeta_1(\left\| v_{IF}^F \right\|)$ by Assumption 5. Hence, $U_1(t) \in \mathcal{L}_\infty$ from (71), $U_2(t)$ is bounded from (73). Since $L_\omega(t)$ and $R_F^C(\theta_c) \in \mathcal{L}_\infty$, $\bar{L}_\omega(t)$ and $\bar{L}_\omega^{-1}(t)$ are bounded from (58) and (66), respectively. Owing to the fact that $B(t)$ is invertible and bounded from (59) and $\bar{B}(t)$ from (66), and then $\bar{U}(t)$ in (61) is bounded. This yields that $u(t)$, $\omega_{IF}^F(t)$, and $\dot{\theta}_c(t)$ are all bounded. $\dot{\Theta}_{IC}^I(t)$ in (26), $\dot{\Theta}_{IF}^I(t)$ in (2), and $\dot{R}_F^I(\Theta)$ in (3) are all bounded, resulting that the camera velocity $\omega_{IC}^C(t)$ in (24), $\omega_{BC}^F(t)$ in (25), and $\omega_{CCd}^C(t)$ in (46) are all bounded. Hence, we can make a conclusion that the time derivatives of errors, $\dot{e}_\theta(t)$, $\dot{e}_p(t)$, $\dot{r}_p(t)$, and $\dot{r}(t)$ in (43), (35), (55), and (64), respectively, are all bounded. Finally $\dot{v}_{IF}^F(t)$ in the modeling equaion in (4) is bounded. Therefore we can conclude that all the signals remain bounded in the suggested closed-loop system.

Appendix B
Kinematics of the 3-Link Camera Positioner

The Denavit-Hartenburg values in Table I, II can be used in conjunction with the formula for the rotation matrix from the i^{th} to $(i-1)^{th}$ frame expressed in the coordinate system $i-1$, that is, $R_i^{i-1}(\Theta)$, according to [12]

$$R_i^{i-1} = \begin{bmatrix} \cos\theta_a & -\sin\theta_a\cos\alpha_t & \sin\theta_a\sin\alpha_t \\ \sin\theta_a & \cos\theta_a\cos\alpha_t & -\cos\theta_a\sin\alpha_t \\ 0 & \sin\alpha_t & \cos\alpha_t \end{bmatrix}. \quad (98)$$

Case1:Tilt-Roll Camera Configuration
(camera looking forward)

All of the rotation matrices of the Tilt-Roll camera positioning unit shown in Figure 3 are obtained using Table I and are given by

$$R_1^{B=O_0} = \begin{bmatrix} \cos\theta_t & 0 & -\sin\theta_t \\ \sin\theta_t & 0 & \cos\theta_t \\ 0 & -1 & 0 \end{bmatrix}_{(Tilt)}, \quad (99)$$

$$R_2^1 = \begin{bmatrix} \sin\theta_p & 0 & \cos\theta_p \\ -\cos\theta_p & 0 & \sin\theta_p \\ 0 & -1 & 0 \end{bmatrix}_{(Pan)} \quad (100)$$

$$\text{and } R_{3=C}^2 = \begin{bmatrix} \sin\theta_r & \cos\theta_r & 0 \\ -\cos\theta_r & \sin\theta_r & 0 \\ 0 & 0 & 1 \end{bmatrix}_{(Roll)}. \quad (101)$$

Tilt-Roll Camera Configuration is achieved by locking the pan angle at $\theta_p(t) = 0$. The rotation matrices $R_1^F(\Theta)$ and $R_2^F(\Theta)$ are defined here for computing (10) and (15). The rotation matrix from the first link frame, O_1, to the quadrotor frame, F, is obtained using

$$\begin{aligned} R_1^F &= R_B^F R_1^{B=O_0} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} \cos\theta_t & 0 & -\sin\theta_t \\ \sin\theta_t & 0 & \cos\theta_t \\ 0 & -1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta_t & 0 & -\sin\theta_t \\ 0 & -1 & 0 \\ -\sin\theta_t & 0 & -\cos\theta_t \end{bmatrix} \end{aligned} \quad (102)$$

where the third column in (102) is the vector $z_1(t)$ in the Jacobian matrix $J_C(t)$. Next, the rotation matrix from the second link frame, O_2, to the quadrotor frame, F, is obtained using

$$\begin{aligned} R_2^F &= R_1^F R_2^1 \\ &= \begin{bmatrix} \cos\theta_t & 0 & -\sin\theta_t \\ 0 & -1 & 0 \\ -\sin\theta_t & 0 & -\cos\theta_t \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & \sin\theta_t & \cos\theta_t \\ 1 & 0 & 0 \\ 0 & \cos\theta_t & -\sin\theta_t \end{bmatrix} \end{aligned} \quad (103)$$

where the third column in (103) is the vector $z_2(t)$ in the Jacobian matrix $J_C(t)$.

Case2:Pan-Tilt Camera Configuration
(camera looking downward)

The Denavit-Hartenburg values in Table II are used to obtain the rotation matrices of the Pan-Tilt camera positioner configuration.

Link	d (offset)	a (length)	α_t (twist)	θ_a (angle)
1	0	0	$-90°$	$\theta_t - 90°$
2	0	0	$-90°$	$\theta_p - 90°$
3	0	0	$0°$	$\theta_r - 90°$

TABLE II
Denavit-Hartenburg Table for a Pan-Tilt Camera Positioner

The rotation matrices of the Pan-Tilt camera positioning configuration shown in Figure 3 are given by

$$R_1^{B=O_0} = \begin{bmatrix} \sin\theta_t & 0 & \cos\theta_t \\ -\cos\theta_t & 0 & \sin\theta_t \\ 0 & -1 & 0 \end{bmatrix}_{(Tilt)} \quad (104)$$

$$R_2^1 = \begin{bmatrix} \sin\theta_p & 0 & \cos\theta_p \\ -\cos\theta_p & 0 & \sin\theta_p \\ 0 & -1 & 0 \end{bmatrix}_{(Pan)} \quad (105)$$

$$\text{and } R_{3=C}^2 = \begin{bmatrix} \sin\theta_r & \cos\theta_r & 0 \\ -\cos\theta_r & \sin\theta_r & 0 \\ 0 & 0 & 1 \end{bmatrix}_{(Roll)}. \quad (106)$$

Pan-Tilt camera configuration is achieved by locking the roll angle at $\theta_r(t) = 0°$ in (106), then we have

$$R_3^2 = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}_{(\theta_r = 0°)}. \quad (107)$$

The rotation matrix $R_B^F(\Theta)$ has same matrix in (8). The rotation matrix from the quadrotor frame (F) to first link frame O_1 is obtained by multiplying the

$$\begin{aligned} R_1^F &= R_B^F R_1^{B=O_0} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} \sin\theta_t & 0 & \cos\theta_t \\ -\cos\theta_t & 0 & \sin\theta_t \\ 0 & -1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} \sin\theta_t & 0 & \cos\theta_t \\ 0 & -1 & 0 \\ \cos\theta_t & 0 & -\sin\theta_t \end{bmatrix} \end{aligned} \quad (108)$$

where the third column in (108) is the second vector in the Jacobian matrix $J_C(t)$. Next, the rotation matrix from the quadrotor frame through the second link yields

$$\begin{aligned} &R_2^F = R_1^F R_2^1 \\ &= \begin{bmatrix} \sin\theta_t & 0 & \cos\theta_t \\ 0 & -1 & 0 \\ \cos\theta_t & 0 & -\sin\theta_t \end{bmatrix} \begin{bmatrix} \sin\theta_p & 0 & \cos\theta_p \\ -\cos\theta_p & 0 & \sin\theta_p \\ 0 & -1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} \sin\theta_t\sin\theta_p & -\cos\theta_t & \sin\theta_t\cos\theta_p \\ \cos\theta_p & 0 & -\sin\theta_p \\ \cos\theta_t\sin\theta_p & \sin\theta_t & \cos\theta_t\cos\theta_p \end{bmatrix} \end{aligned} \quad (109)$$

where the third column is the third vector in the Jacobian matrix for Pan-Tilt manipulator. Then, we can obtain

the total rotation matrix by combining all those matrices from quadrotor through the 3rd link to yield

$$R_C^F = \begin{bmatrix} \cos\theta_t & \sin\theta_t \sin\theta_p & \sin\theta_t \cos\theta_p \\ 0 & \cos\theta_p & -\sin\theta_p \\ -\sin\theta_t & \cos\theta_t \sin\theta_p & \cos\theta_t \cos\theta_p \end{bmatrix}. \quad (110)$$

The Jacobian matrix of the Pan-Tilt camera manipulator, $J_C'(\Theta) \in \mathbb{R}^{3\times 3}$, can be represented

$$J_C^{'} = \begin{bmatrix} 0 & \cos\theta_t & \sin\theta_t \cos\theta_p \\ 1 & 0 & -\sin\theta_p \\ 0 & -\sin\theta_t & \cos\theta_t \cos\theta_p \end{bmatrix}, \quad (111)$$

$$\theta_C^{'} = \begin{bmatrix} \theta_t \\ \theta_p \\ \theta_r \end{bmatrix}^{\mathsf{T}} \text{ where } \theta_r(t) = 0^\circ. \quad (112)$$

To simplify the use of the Jacobian matrix, it can be noted that $\dot{\theta}_r(t) = 0^\circ$ and hence a new reduced position joint angle vector can be introduced as

$$\dot{\theta}_C = \begin{bmatrix} \dot{\theta}_t \\ \dot{\theta}_p \end{bmatrix} \in \mathbb{R}^2 \text{ and } \theta_C = \begin{bmatrix} \theta_t \\ \theta_p \end{bmatrix} \in \mathbb{R}^2 \quad (113)$$

along with a reduced Jacobian matrix, $J_C(\Theta) \in \mathbb{R}^{3\times 2}$, defined as

$$J_C = \begin{bmatrix} 0 & \cos\theta_t \\ 1 & 0 \\ 0 & -\sin\theta_t \end{bmatrix}. \quad (114)$$

Appendix C
Simulation Results

Simulation results are obtained as

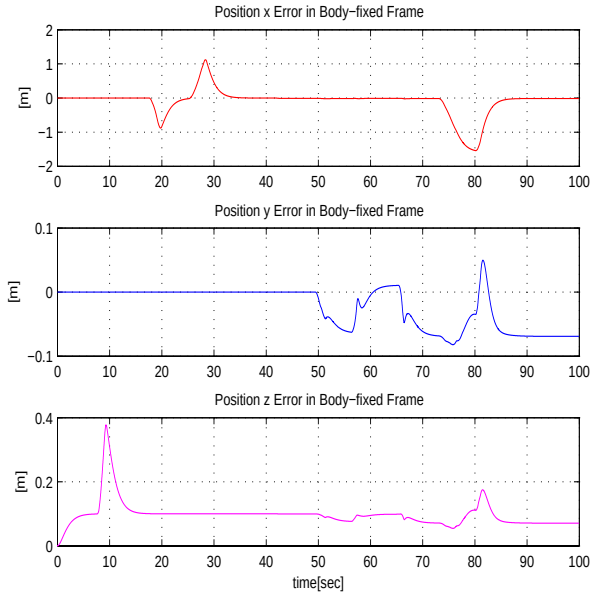


Fig. 7. position tracking errors

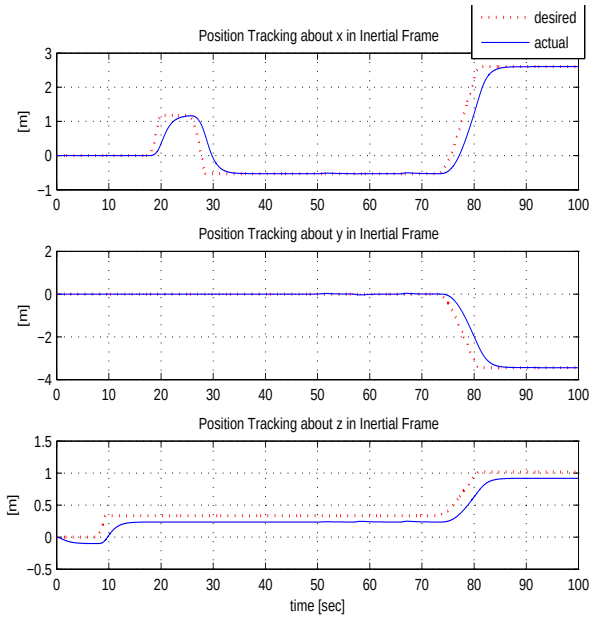


Fig. 8. position tracking

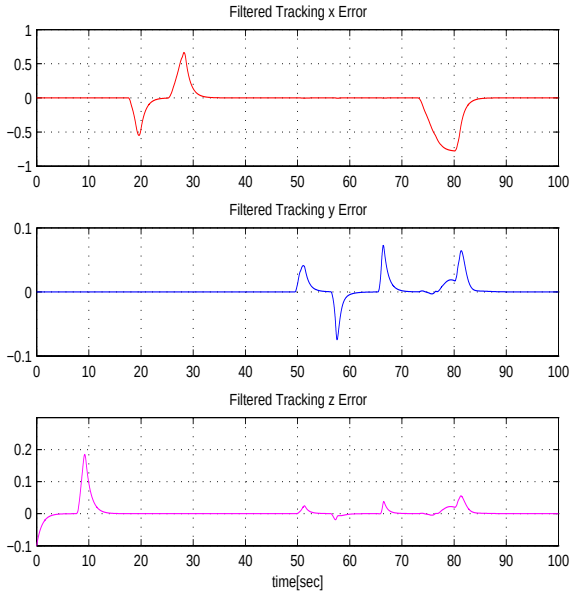


Fig. 9. filtered tracking errors

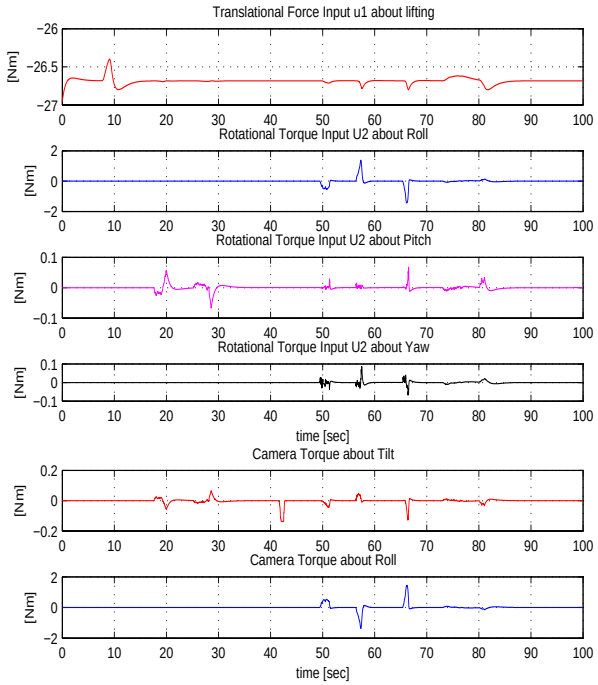


Fig. 10. Control Inputs

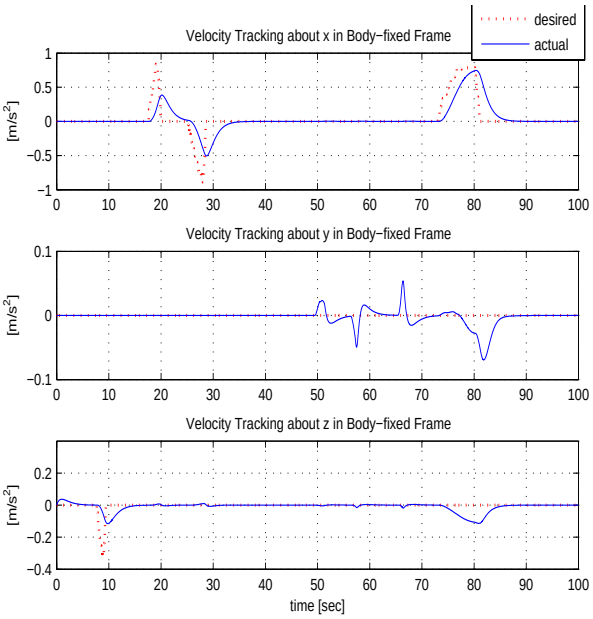


Fig. 12. velocity tracking

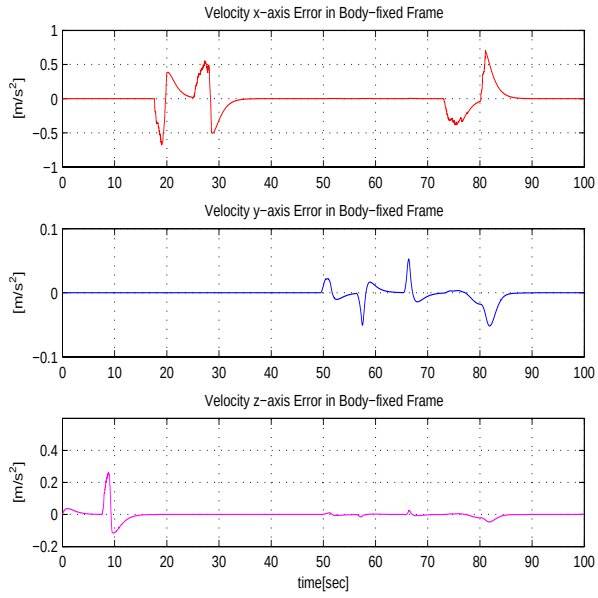


Fig. 11. velocity tracking errors

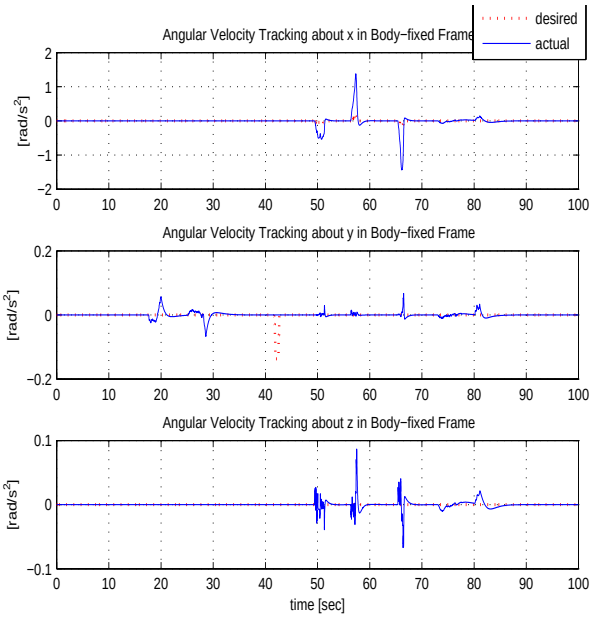


Fig. 13. angular velocity tracking

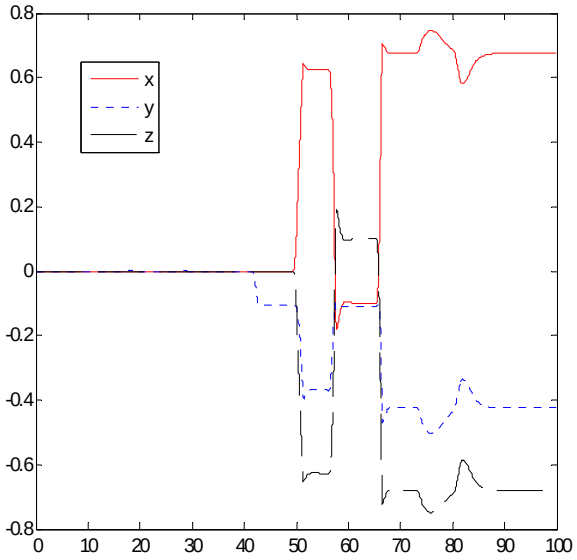


Fig. 14. eTheta

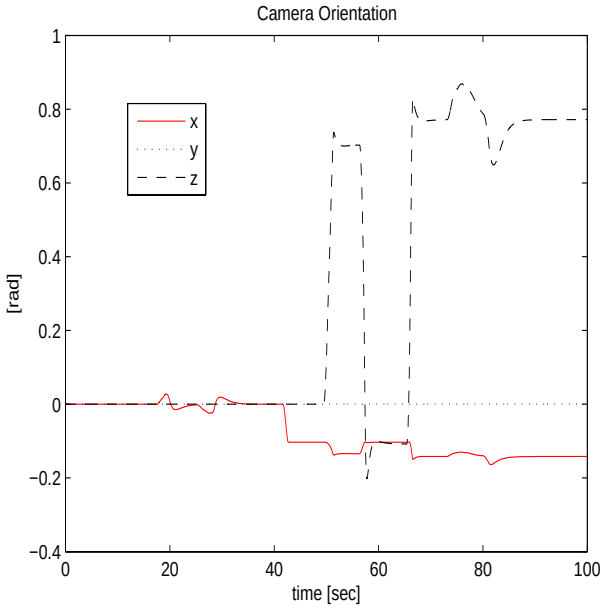


Fig. 15. Camera Orientation

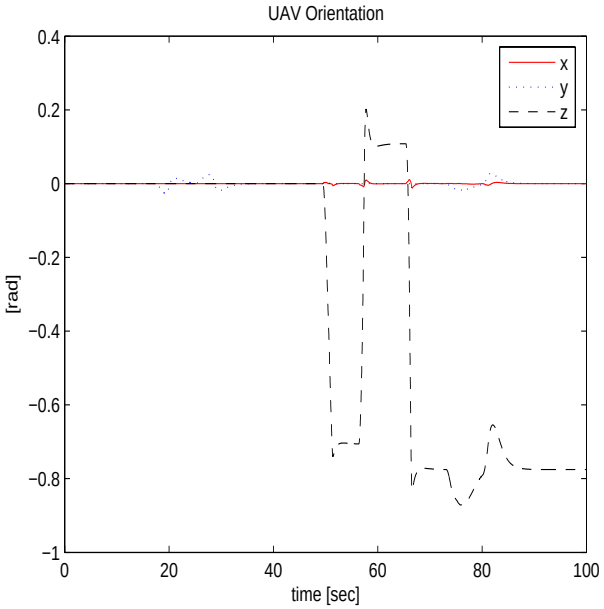


Fig. 16. UAV orientation