Toward Image Based Visual Servoing for Aerial Grasping and Perching

Justin Thomas, Giuseppe Loianno, Koushil Sreenath, and Vijay Kumar

Abstract— **This paper addresses the dynamics, control, planning, and visual servoing for micro aerial vehicles to perform high-speed aerial grasping tasks. We draw inspiration from agile, fast-moving birds, such as raptors, that detect, locate, and execute high-speed swoop maneuvers to capture prey. Since these grasping maneuvers are predominantly in the sagittal plane, we consider the planar system and present mathematical models and algorithms for motion planning and control, required to incorporate similar capabilities in quadrotors equipped with a monocular camera. In particular, we develop a dynamical model directly in the image space, show that this is a differentially-flat system with the image features serving as flat outputs, outline a method for generating trajectories directly in the image feature space, develop a geometric visual controller that considers the second order dynamics (in contrast to most visual servoing controllers that assume first order dynamics), and present validation of our methods through both simulations and experiments.**

I. Introduction

The dexterity and adaptability of living creatures far exceeds what we see in modern robots but provides great inspiration for avenues of research. For example, birds of prey can not only cover great distances by flight, but they are also excellent hunters and are able to perch and walk. Some raptors exhibit great speed and agility, and they use visual target recognition and perception-action loops to swoop down to grasp their prey [1]. For example, see Fig. 1 for still images of a Red Kite swooping to capture a target.

Such capacities would be advantageous for robots in aerial manipulation tasks. For example, placing sensors quickly, perching to save energy, and dynamic grasping to acquire and transport materials or other robots, are a few tasks that increase the usefulness of aerial robots and require similar capabilities to those of raptors. Aerial robots, however, do not yet have a comparable rich set of capabilities, and they are limited partially by low strength/weight actuators, heavy materials, and batteries with low energy density and low specific power. In addition, our poor understanding of the perception-action loops required for agile flight and manipulation remains a limiting factor. For grasping or perching, the robot must be able to detect the object of interest and subsequently use visual feedback to control the robot's motion. To maintain agility, the robot must also accomplish this task with a minimal sensor payload and consideration of the dynamics of the system [2], [3].

Fig. 1. A Red Kite swoops down and uses visual feedback to approach, grasp, and retrieve food on the ground [8].

In scenarios like this, a monocular camera is an inexpensive and versatile sensor of choice, especially when combined with an Inertial Measurement Unit (IMU) [4], [5]. Such applications requiring control feedback using a single camera motivate either Position Based Visual Servoing (PBVS) or Image Based Visual Servoing (IBVS) [6]. PBVS requires an explicit estimation of the pose of the robot in the inertial frame while IBVS acts directly using feedback from the image coordinates. In particular, a single monocular camera is sufficient for visual servoing when there is some known geometry or structure in the environment. Further, visual servoing can even be effective in an unknown environment while remaining insensitive to camera calibration [7].

Our goal is to ascribe to aerial vehicles the ability to fly above, grasp, or perch on a target using vision. The platform considered is a quadrotor micro aerial vehicle (MAV) which is similar to a helicopter, but has four rotors [9]. The quadrotor platform is appealing because of its mechanical simplicity, agility, and well understood dynamics [10].

Despite the fact that it is underactuated, it is possible to design controllers that will guarantee convergence from almost any point on $SE(3)$, the Euclidean motion group. In our group's previous work, similar controllers have also been derived for a quadrotor carrying a cable-suspended payload [10]. However, both of these approaches require full knowledge of the state. Therefore, our goal is to use similar approaches with the dynamics of the system directly in the image plane (rather than in the Cartesian space) to develop an IBVS controller based on visual features of a cylinder. We will also demonstrate a method to generate dynamically-feasible trajectories in the image plane.

There are many excellent tutorials on visual servoing [11], [6], [12], [13]. However, most approaches are limited to first-order or fully-actuated systems. For example, Taylor and Ostrowski demonstrated robustness to camera calibration, but only considered a first-order system [7]. Cowan *et al.* proved stability for second order systems, but assumed a fully actuated system [14]. More recently, Hamel and Mahoney

*We gratefully acknowledge the support of ARL grant W911NF-08-2-0004 and ONR grants N00014-07-1-0829 and N00014-08-1-0696.

J. Thomas, G. Loianno, and V. Kumar are with the GRASP Lab, University of Pennsylvania, Philadelphia, PA, USA {jut, loiannog, kumar}@seas.upenn.edu

K. Sreenath is with the Depts. of Mechanical Engineering and Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA koushils@cmu.edu

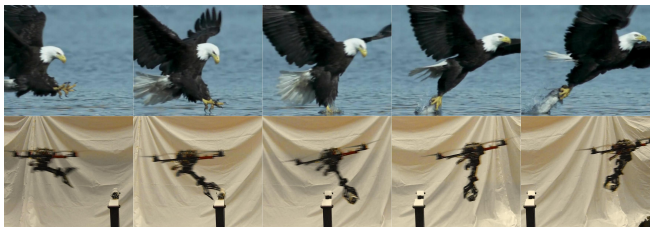

Fig. 2. Top: A bald eagle snatches a fish from the water [18]. Bottom: A quadrotor dynamically grasps at 3 m/s using a custom gripper [2].

leveraged a spherical camera model and utilized backstepping to design non-linear controllers for a specific class of underactuated second-order systems [15], [16]. As is typical in backstepping, however, it is necessary to assume that the inner control loops are significantly faster than the outer ones. Some preliminary efforts have been made to enable autonomous landing, but require an estimate of velocity in the inertial frame using an external motion capture system [17]. Therefore, there is a lack of IBVS controllers which can handle the dynamic motion required for aggressive grasping and perching.

In this paper, we build upon existing IBVS literature, generalizing from the typical first-order fully actuated system to a higher-order underactuated system, and we develop IBVS controllers for dynamic vision-based grasping and perching. In our previous work, we demonstrated a proof-of-concept bio-inspired gripper which, when attached to a quadrotor, enabled dynamic grasping of stationary objects as displayed in Fig. 2 while moving at 3 m/s (or 9 body lengths / s) [2]. However, this work required a motion capture system [19] and therefore restricted such aggressive maneuvers to a structured environment.

The primary contribution of this paper is to enable high-speed grasping maneuvers by developing a dynamical model directly in the image space, showing that this is a differentially-flat system with the image features serving as flat outputs, developing a geometric visual controller that considers the second order dynamics (in contrast to most visual servoing controllers that assume first order dynamics), and presenting validation of our methods through both simulations and experiments[1].

The rest of the paper is structured as follows. In Section II, a mapping from Cartesian space to the image space is presented. Section III develops a dynamical model in the image space, while Section IV establishes that the system is differentially-flat, with the visual features being the flat outputs, and presents a trajectory generation method. Section V develops an IBVS controller for the full dynamical system using geometric techniques. Finally, Sections VI and VII present simulation and experimental results, respectively.

[1]It must be noted that grasping maneuvers are predominanty in the sagittal plane and thus our developed models and algorithms for motion planning and control are based on a planar model ($x-z$ plane). However, since the experimental system is 3D, we will use a Vicon-based motion capture system to ensure stability for the yaw and the y-axis dynamics. The $x-z$ dynamics will be stabilized thorugh our developed IBVS controller.

II. Vision

In this section, we present an overview of the vision system, outline the camera model, and derive the geometric constraints on the cylinder detection in the image plane.

A. Problem Formulation

The problem is formulated in the sagittal plane, which is the dominant plane of actuation for aerial grasping maneuvers in nature and enables high-speed dynamic grasping for aerial robots [2]. This also allows us to simplify the problem. When considering the sagittal plane, two image features are sufficient to establish a relationship between the vision system and the pose of the robot. These features could be any two known points; however, it is not always possible to have such a structured environment in practice. Conveniently, cylinders with known (or estimated) radii are fairly common and provide sufficient structure for visual servoing and grasping.

We will use the following nomenclature. Let T be the homogeneous transformation matrix from the camera frame to the world frame, f_i denote a focal length in the i^{th} direction, c_i be the center image pixel in the i^{th} direction, and λ be an arbitrary scaling factor.

B. Camera Model

The camera is modeled using a standard pinhole perspective camera model so that a generic point in the world, $\begin{bmatrix} X & Y & Z & 1 \end{bmatrix}^T$, is projected onto the image plane, $\begin{bmatrix} x' & y' & 1 \end{bmatrix}^T$, according to [20] such that

$$\lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = K P_0 T^{-1} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix},$$

where

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \quad P_0 = \begin{bmatrix} \mathcal{I}_{3\times 3} & \mathbf{0}_{3\times 1} \end{bmatrix}.$$

From here on, we will use the calibrated image coordinates,

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K^{-1} \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix},$$

which are equivalent to the transformation and projection of points in the world to an image plane with unity focal length and a centered coordinate system.

C. Geometry

Let the image features be the points whose rays are tangent to the cylinder and lie in the vertical plane. In contrast to typical visual servoing approaches, these points are now a function of the position of the robot. Therefore, we cannot use the standard image Jacobian as in [12], which assumes the target points are stationary in the inertial frame.

We now formulate the mapping between the image features and the robot pose. Let $\mathbf{r}_q = (x_q, z_q)^T \in \mathbb{R}^2$ denote the

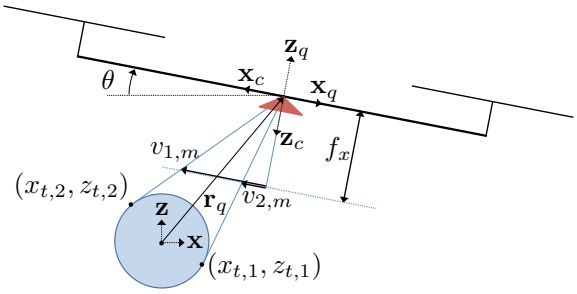

Fig. 3. We assume, without loss of generality, that the target is located at the origin and the quadrotor is located at $(x_q,\, z_q)$. The focal length of the camera, f_x, defines the location of the image plane relative to the quadrotor and the image coordinates are given by v_1 and v_2. The optical ray tangent to the target intersects the target at $(x_t,\, z_t)$. The coordinate system of the camera is indicated by $\mathbf{x}_c$ and $\mathbf{z}_c$.

position of the quadrotor in the inertial frame with the target cylinder at the origin, as shown in Fig. 3. Let R_t denote the radius of the target cylinder, and $\mathbf{r}_{t,i} \in \mathbb{R}^2$ be a point on the circumference that has a tangent passing through the focal point. With the camera at the same position as the quadrotor, we have two geometric constraints,

$$\|\mathbf{r}_t\|_2 = R_t$$
$$\|\mathbf{r}_q\|_2^2 = \|\mathbf{r}_q - \mathbf{r}_t\|_2^2 + R_t^2.$$

Knowing the radius of the cylinder, R_t, these equations have two solutions which represent the two tangent points,

$$\mathbf{r}_{t,i} = \frac{R_t^2}{\|\mathbf{r}_q\|^2}\left(\begin{bmatrix} x_q \\ z_q \end{bmatrix} \pm \begin{bmatrix} -z_q \\ x_q \end{bmatrix}\sqrt{\frac{\|\mathbf{r}_q\|^2}{R_t^2} - 1}\right). \quad (1)$$

Unfortunately, the features in the image plane are coupled with the attitude, and if not compensated, would not allow for the desired attitude-decoupled mapping between the state of the robot and the image features. Therefore, we map the calibrated image coordinates to coordinates on a virtual level image plane similar to [21] by rotating the camera coordinate system to a virtual frame where $\theta = 0$. Then, the problem can be formulated in the rotated coordinate system where the virtual image plane is level. The virtual calibrated coordinates of the features can then be computed using (1) and

$$\lambda \begin{bmatrix} v_i \\ 0 \\ 1 \end{bmatrix} = P_0 T^{-1} \begin{bmatrix} x_{t,i} \\ 0 \\ z_{t,i} \\ 1 \end{bmatrix} \quad (2)$$

with the appropriate transformation $T \in SE(3)$. The virtual coordinates, $\mathbf{v} = [v_1,\ v_2]^T$, in (2) provide two equations which can be solved to determine the robot and camera position as a function of the virtual image coordinates.

We also define the space $S = \{\mathbf{r}_q \in \mathbb{R}^2 \mid 2R_t \leq \|\mathbf{r}_q\| \leq B_r, z_q > 0\}$, such that the quadrotor's position is bounded below by $2R_t$ and bounded above by B_r, and the quadrotor is always above the horizontal plane. Then, there exists $V \subset \mathbb{R}^2$ and a smooth global diffeomorphism $\Gamma : S \longrightarrow V$ such that

$$\mathbf{v} = \frac{f_x}{z_q^2 - R_t^2}\begin{bmatrix} x_q z_q + R_t^2\sqrt{\frac{\|\mathbf{r}_q\|^2}{R_t^2} - 1} \\ x_q z_q - R_t^2\sqrt{\frac{\|\mathbf{r}_q\|^2}{R_t^2} - 1} \end{bmatrix} \equiv \Gamma(\mathbf{r}_q), \quad (3)$$

$$\dot{\mathbf{v}} = \frac{d\Gamma(\mathbf{r}_q)}{dt} = \frac{\partial}{\partial \dot{\mathbf{r}}_q}\left(\frac{d\Gamma(\mathbf{r}_q)}{dt}\right)\dot{\mathbf{r}}_q \equiv J\dot{\mathbf{r}}_q,$$

where J is the image Jacobian [22]. Note that J can be expressed as a function of either the image coordinates or the position of the robot by using (3) and the fact that Γ is invertible. Having established a mapping between the Cartesian coordinates and the image coordinates, we will next develop a dynamic model of the quadrotor system directly in the image coordinates.

III. Dynamics

The dynamics of this quadrotor system are well known in literature. For simplicity, we restrict the robot to the vertical $(x - z)$ plane as shown in Fig. 4 and we assume the gripper is massless (See [23] for the complete 3-D dynamic model). We define

$$\mathbf{r}_q = \begin{bmatrix} x_q \\ z_q \end{bmatrix}, \quad \mathbf{w}_q = \begin{bmatrix} \mathbf{r}_q \\ \theta \end{bmatrix}$$

where $\mathbf{r}_q$ is the position of the quadrotor and θ is the pitch angle. Then, the dynamics in the inertial frame take the form

$$D\ddot{\mathbf{w}} + C\dot{\mathbf{w}} + \mathbf{G} = \mathbf{F} \quad (4)$$

where $D \in \mathbb{R}^{3\times3}$ is a diagonal inertial tensor because the robot frame is aligned with the principal axes of the inertia. In this case, centripetal and Coriolis terms, $C \in \mathbb{R}^{3\times3}$, are zero. Gravity appears in $\mathbf{G} \in \mathbb{R}^{3\times1}$, and $\mathbf{F} \in \mathbb{R}^{3\times1}$ is

$$\mathbf{F} = \begin{bmatrix} fR\mathbf{e}_2 \\ M \end{bmatrix}$$

where $R \in SO(2)$, $f \in \mathbb{R}$ is the total thrust, $\mathbf{e}_2 = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$, and M is the pitch moment generated from the difference of thrusts between the front and rear rotors as depicted in Fig. 4. Since the system has three degrees of freedom, given by $\mathbf{w}_q$, and only two control inputs that appear in $\mathbf{F}$, the system is underactuated.

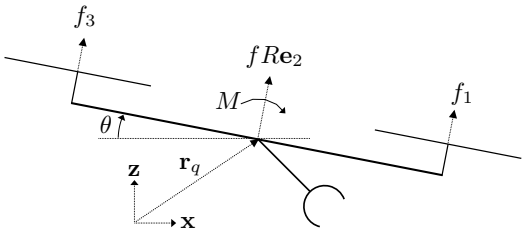

Fig. 4. The total thrust is $f = \sum_{i=1}^{4} f_i$ and is in the direction of $R\mathbf{e}_2$. The moment, M, is the result of the thrust difference between f_3 and f_1.

Now, $\dot{\mathbf{r}}_q$ and $\ddot{\mathbf{r}}_q$ can be expressed as functions of the image coordinates using the inverse of the image Jacobian, J. Then, the dynamics in (4) can be expressed in terms of the image coordinates

$$D\begin{bmatrix} J^{-1}\ddot{\mathbf{v}} - J^{-1}\dot{J}J^{-1}\dot{\mathbf{v}} \\ \ddot{\theta} \end{bmatrix} + C\begin{bmatrix} J^{-1}\dot{\mathbf{v}} \\ \dot{\theta} \end{bmatrix} + \mathbf{G} = \mathbf{F}$$

which simplifies to:

$$\ddot{\mathbf{v}} = \frac{1}{m}J\left[fR\mathbf{e}_2 - \mathbf{G}_A\right] + \dot{J}J^{-1}\dot{\mathbf{v}} \quad (5)$$

$$D_3\ddot{\theta} = M \quad (6)$$

where G_A denotes the upper left 2×2 block of G, D_3 is the $\{3,3\}$ element of D, and m is the mass of the robot. Equation (5) presents the translational dynamics directly in the image coordinates. Next, we will demonstrate that these coordinates form a set of flat outputs for the system, enabling trajectory design directly in the image space.

IV. Differential Flatness and Planning

In this section, we formulate the trajectory planning problem in the image plane and compute trajectories that satisfy the dynamic constraints (i.e. are dynamically feasible) in (5) and (6). The central idea relies on the differential flatness (see [24] for definition) property of our system. We will show that there is a set of so-called "flat outputs" such that there is a diffeomorphism from the vector of state variables and inputs to the vector of flat outputs and its derivatives. In practice, this means that dynamically feasible trajectories can be generated by considering sufficiently smooth (the order is defined by the map) trajectories in the space of flat outputs. In this paper, we show that the image coordinates, $\mathbf{v} \in V \subset \mathbb{R}^2$, serve as flat outputs.

First, there exists a diffeomorphism between the image coordinates and the position of the robot, namely Γ as defined in (3). From (5), we get:

$$fR\mathbf{e}_2 = mJ^{-1}\left(\ddot{\mathbf{v}} - \dot{J}J^{-1}\dot{\mathbf{v}}\right) + \mathbf{G}_A,$$

and

$$f = \|\mathbf{F}_A\|, \quad \theta = \arctan\left(\frac{F_1}{F_2}\right).$$

where $\mathbf{F}_A = mJ^{-1}\left(\ddot{\mathbf{v}} - \dot{J}J^{-1}\dot{\mathbf{v}}\right) + \mathbf{G}_A$. Further,

$$\dot{f} = \mathbf{e}_2^T R^T \dot{\mathbf{F}}_A, \;\; \dot{\theta} = \frac{1}{f}\mathbf{e}_1^T R^T \dot{\mathbf{F}}_A.$$

Since f appears in the denominator, we require that $f > 0$ so that the thrust is always positive. From (6) and one more derivative, we obtain:

$$M = D_3 \frac{1}{f}\left(\mathbf{e}_1^T R^T \ddot{\mathbf{F}}_A - 2\dot{f}\dot{\theta}\right). \tag{7}$$

Thus, all state variables and inputs can be written as functions of the image coordinates and their derivatives. Since we require the 4^{th} derivative of the image coordinates (see (7)), the planned trajectories in the image plane must be at least C^4 continuous.

Following previous work [25], [2], since the input M is an algebraic function of the fourth derivative (snap), it is natural to plan smooth trajectories that minimize the snap of the trajectory using the cost functional,

$$\mathcal{J}_i = \int_{t_0}^{t_f} \left\| v_i^{(4)}(t) \right\|^2 dt.$$

Choosing a polynomial basis allows the minimization to be formulated as a Quadratic Program (QP) [25]. Further, equality constraints can be enforced and can be determined by desired robot positions (or velocities) using the Γ map or by previously measured image coordinates. Finally, all constraints on attitude and the field of view must be written as inequalities in the virtual image plane. Having obtained a trajectory in the image space, we next develop a visual controller that will track this trajectory.

V. Control

A. Attitude Controller

First, let $R_d \in SO(2)$ denote the desired rotation matrix defined by a desired attitude, θ_d. Then, we define attitude errors

$$e_R = \frac{1}{2}\left(R_d^T R - R^T R_d\right)^\vee = \sin(\theta - \theta_d)$$
$$e_\Omega = \Omega - R^T R_d \Omega_d = \dot{\theta} - \dot{\theta}_d$$

where $^\vee$ is the "vee" map defined in [26]. These errors are similar to [26] but simplified for the planar case. Also, we define a configuration error function as

$$\Psi(R, R_d) = \frac{1}{2}\text{tr}\left[I - R_d^T R\right].$$

The attitude controller is then given as below.

***Proposition** 1:* [26, Prop. 1] (Exponential Stability of Attitude Controlled Flight Mode) Consider the control moment defined as

$$M = -K_R e_R - K_\Omega e_\Omega + D_3 \ddot{\theta}_d,$$

where K_R and K_Ω are positive scalars. Further, suppose the initial conditions satisfy

$$\Psi(R(0), R_d(0)) < 2,$$
$$\|e_\Omega(0)\|^2 < \frac{2}{D_3} k_R \left(2 - \Psi(R(0), R_d(0))\right).$$

Then, $(e_R, e_\Omega) = (0, 0)$ is exponentially stable for the closed-loop system.

Proof: Follows from [26, Prop. 1]. See `http://www.jtwebs.net/ICRA-2014/` for details. ■

B. Position Control

Let errors in the image plane be defined by

$$\mathbf{e}_v = \mathbf{v} - \mathbf{v}_d.$$

Then, using (5), the image space error dynamics are

$$m\ddot{\mathbf{e}}_v = fJR\mathbf{e}_2 - J\mathbf{G}_A + m\dot{J}J^{-1}\dot{\mathbf{v}} - m\ddot{\mathbf{v}}_d.$$

The visual servoing controller in then given as below.

***Proposition** 2:* (Exponential Stability of Visual Feature Controlled Flight Mode) Consider the total thrust component along the current body frame vertical axis defined by

$$f = \mathbf{A} \cdot R\mathbf{e}_2.$$

where $\mathbf{A} = \mathbf{G}_A + mJ^{-1}\left[-K_p\mathbf{e}_v - K_d\dot{\mathbf{e}}_v + \ddot{\mathbf{v}}_d\right]$, and the commanded attitude given by

$$R_c\mathbf{e}_2 = \frac{\mathbf{A}}{\|\mathbf{A}\|}.$$

Then, the zero equilibrium $(\mathbf{e}_v, \dot{\mathbf{e}}_v, e_R, e_\Omega) = (\mathbf{0}, \mathbf{0}, 0, 0)$ is locally exponentially stable.

Proof: See `http://www.jtwebs.net/ICRA-2014/` for details. ■

VI. Simulation Results

Using the trajectory generation method outlined in Section IV, we can generate sample trajectories directly in the image coordinates, representing a swooping maneuver. It is reasonable to specify a limit on the attitude, which enables the incorporation of linear visibility constraints, rather than requiring non-linear visibility constraints when planning in the Cartesian space. A sample trajectory is shown in Fig. 5 (top), where the boundary conditions and intermediate waypoint were computed using Γ, and with the derivatives in the intermediate waypoint left unconstrained.

Next, using the generated desired trajectory in the image plane, the controller from Section V is simulated on the dynamic model given by (5)-(6). The simulation is started with an initial image coordinate error of $0.10m$, and the resulting trajectory and error are plotted in Fig. 5.

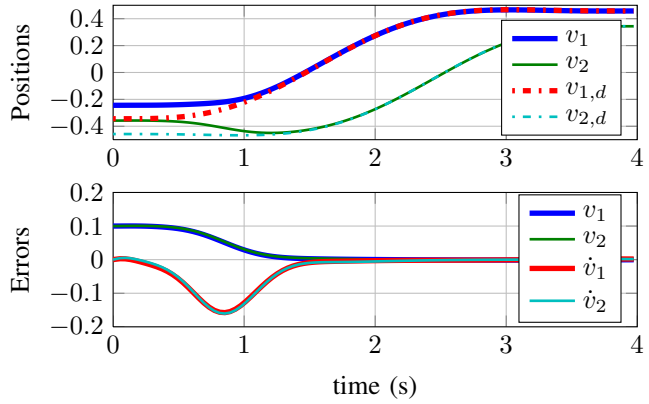

Fig. 5. A sample trajectory in simulation. The simulated image coordinates, v_i, and the desired coordinates, $v_{i,d}$, are in the top graph where there is an initial error of $0.1m$ in each coordinate. The feature errors and error velocities are in the bottom graph.

VII. Experimental Validation

A Hummingbird quadrotor equipped with a global shutter Caspa™ VL camera and Computer on Module from Gumstix [27] is used for experiments to fly down and grasp a cylinder object. As mentioned earlier, grasping maneuvers are predominantly in the sagittal plane and thus our developed models and algorithms for motion planning and control are planar. Since the experiments are on a 3D system, an external Vicon-based motion capture system is used to stabilize the yaw and lateral dynamics while our IBVS controller stabilizes motion in the sagittal plane.

Visual detection and tracking of the cylinder object runs onboard the robot, is based on blob tracking using Freeman chain coding, and is obtained using the C++ Visp library [28]. When the object is in the image and $\mathbf{r}_q \in S$, the measured image points from the camera are mapped to the virtual image plane using feedback from the IMU and the transformation shown in Fig. 6, which is mathematically equivalent to

$$v_i = \tan\left(\arctan\left(v_{i,m}\right) + \theta\right)$$

where $v_{i,m}$ is the boundary of the cylinder in the actual calibrated image. The points in the virtual plane are filtered

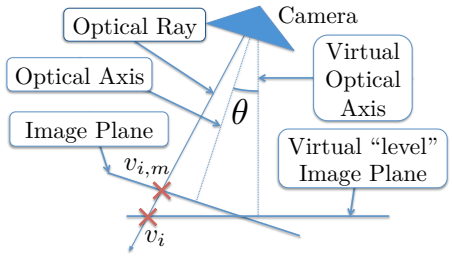

Fig. 6. The measured image feature points, $v_{i,m}$, which are affected by θ, are projected onto a virtual level image plane to decouple the motion from the attitude of the robot and determine the coordinates v_i.

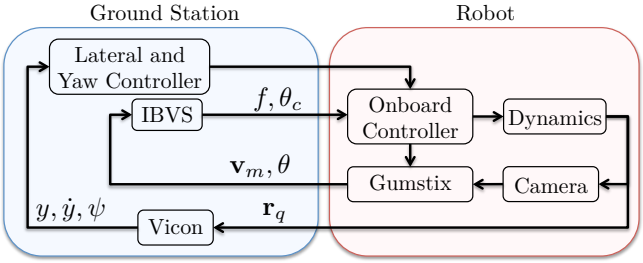

Fig. 7. A camera captures images of the cylinder, which are sent to the Gumstix Overo Computer on Module (COM). The images are processed at 65 Hz using blob tracking; the boundaries of the cylinder are undistored, calibrated, and sent back to a ground station along with the pitch as measured from the IMU. From here, the ground station maps the points to the virtual plane and computes desired control inputs using the IBVS controller. In parallel, Vicon feedback is used to close the loop on the lateral $(y, \dot{y})$ and yaw (ψ) dimensions of the robot. Then, the desired attitude is sent to the onboard controller, which uses the IMU to control the attitude at 1 kHz.

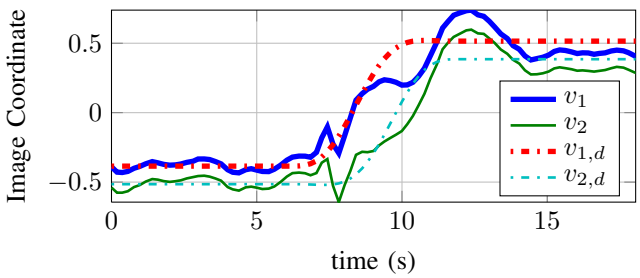

Fig. 8. Experimental results of the feature coordinates in the virtual plane for a "swooping" trajectory. The feature coordinates are denoted by v_i and the desired trajectory is given by $v_{i,d}$.

to improve the estimate of the image features and their derivatives to compute J and $\dot{J}$. A block diagram of the system is shown in Fig. 7.

The stability of the controller was demonstrated through several experiments including hovering, vertical trajectories, "swooping" trajectories, and hovering above a moving cylinder. Here we present a "swooping" trajectory, which includes some components of the mentioned trajectories. See Fig. 8 for the planned and actual trajectories in the virtual image plane, Fig. 9 for the corresponding estimated and actual position in the inertial frame, and Fig. 10 for a sequence of still images from a sample experiment. The reader can observe the other trajectories in the attached video.

The aggression of our trajectories is limited beause of (i) slow feedback due to limited onboard sensing and computation, (ii) a limited optical resolution, and (iii) a small optical field of view.

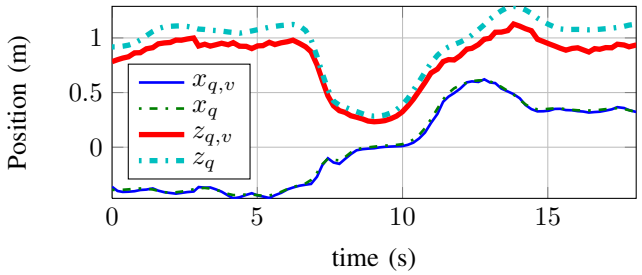

Fig. 9. Positions in the inertial frame for the experiment in Fig. 8. The vision estimates of the position (using Γ) are denoted by the "v" subscript. The ground truth only has the "q" subscript.

Fig. 10. Still images from a sample "swooping" trajectory using the vision-based controller developed in this paper. The background has been washed out slightly to improve visibility.

VIII. Conclusion

This paper demonstrates a first step towards autonomous dynamic grasping and manipulation for micro aerial vehicles in unstructured environments. In particular, we considered a quadrotor system equipped with a monocular camera and formulated the dynamics of the underactuated system directly in the virtual image plane. The system was demonstrated to be differential flat, with the image coordinates being the set of flat outputs. We presented a trajectory generation method which guarantees dynamic feasibility and enables incorporating visual constraints as linear constraints. We developed a non-linear vision-based controller for trajectory tracking in the image space, presented a proof of stability, and provided validation of the controller both in simulation and in experimentation on a quadrotor. A next step would be to generalize the control law to full three dimensions and consider the yaw of the robot by using image moments to detect the primary axis of the cylinder.

Acknowledgment

The authors would like to thank Kartik Mohta for his expertise and assistance with the vision hardware and software.

References

[1] Norma Venable. *Birds of Prey*. West Virginia University Extension Service, 1996.

[2] Justin Thomas, Joe Polin, Koushil Sreenath, and Vijay Kumar. Avian-Inspired Grasping for Quadrotor Micro UAVs. In *International Design and Engineering Technical Conferences & Computers and Information in Engineering Conference (IDETC/CIE)*, Portland, 2013.

[3] Matko Orsag, Christopher Korpela, and Paul Oh. Modeling and Control of MM-UAV: Mobile Manipulating Unmanned Aerial Vehicle. *Journal of Intelligent & Robotic Systems*, 69(1-4):227–240, August 2012.

[4] Anastasios I Mourikis, Stergios I Roumeliotis, and Joel W Burdick. SC-KF Mobile Robot Localization: A Stochastic Cloning Kalman Filter for Processing Relative-State Measurements. *IEEE Transactions on Robotics*, 23(4):717–730, August 2007.

[5] Stephan Weiss, Davide Scaramuzza, and Roland Siegwart. Monocular-SLAM-based navigation for autonomous micro helicopters in GPS-denied environments. *Journal of Field Robotics*, 28(6):854–874, November 2011.

[6] S. Hutchinson, G.D. Hager, and P.I. Corke. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12(5):651–670, 1996.

[7] C.J. Taylor and J.P. Ostrowski. Robust vision-based pose control. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, volume 3, pages 2734–2740. IEEE, 2000.

[8] The Slow Mo Guys. "Red Kites in Slow Motion", http://youtu.be/AYOx-iCMZhk.

[9] "Ascending Technologies, GmbH," http://www.asctec.de.

[10] Koushil Sreenath, Taeyoung Lee, and Vijay Kumar. Geometric Control and Differential Flatness of a Quadrotor UAV with a Cable-Suspended Load. In *IEEE Conference on Decision and Control (CDC)*, pages 2269–2274, Florence, Italy, December 2013.

[11] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation*, 8(3):313–326, June 1992.

[12] Francois Chaumette and Seth Hutchinson. Visual servo control. I. Basic approaches. *IEEE Robotics & Automation Magazine*, 13(4):82–90, December 2006.

[13] F. Chaumette and S. Hutchinson. Visual servo control. II. Advanced approaches [Tutorial]. *IEEE Robotics & Automation Magazine*, 14(1):109–118, March 2007.

[14] NJ Cowan, J.D. Weingarten, and D.E. Koditschek. Visual servoing via navigation functions. *IEEE Transactions on Robotics and Automation*, 18(4):521–533, August 2002.

[15] Tarek Hamel and Robert Mahony. Visual Servoing of an Under-Actuated Dynamic Rigid-Body System: An Image-Based Approach. *IEEE Transactions on Robotics & Automation*, 18(2):187–198, 2002.

[16] Tarek Hamel and Robert Mahony. Image based visual servo control for a class of aerial robotic systems. *Automatica*, 43(11):1975–1983, November 2007.

[17] Daewon Lee, Tyler Ryan, and H. Jin. Kim. Autonomous landing of a VTOL UAV on a moving platform using image-based visual servoing. In *2012 IEEE International Conference on Robotics and Automation*, pages 971–976. IEEE, May 2012.

[18] Karen Bass, Brian Leith, Justin Anderson, Peter Bassett, Joe Stevens, Hugh Pearson, and Jeff Turner. Nature's Most Amazing Events. [DVD]. BBC Worldwide Ltd. Programs, 2009.

[19] "Vicon Motion Systems, Inc," http://www.vicon.com.

[20] Yi Ma, Stefano Soatto, Jana Kosecka, and Shankar Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models*. Interdisciplinary Applied Mathematics. Springer, New York, 2004.

[21] Hamed Jabbari, Giuseppe Oriolo, and Hossein Bolandi. Dynamic IBVS control of an underactuated UAV. *2012 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1158–1163, December 2012.

[22] Hong Zhang and J.P. Ostrowski. Visual Servoing with Dynamics: Control of an Unmanned Blimp. In *International Conference on Robotics and Automation*, volume 1, pages 618–623. IEEE, 1999.

[23] Koushil Sreenath, Nathan Michael, and Vijay Kumar. Trajectory Generation and Control of a Quadrotor with a Cable-Suspended Load A Differentially-Flat Hybrid System. In *International Conference on Robotics and Automation*, 2013.

[24] R.M. Murray, M. Rathinam, and Willem Sluis. Differential flatness of mechanical control systems: A catalog of prototype systems. In *ASME International Congress and Exposition*. Citeseer, 1995.

[25] Daniel Mellinger and Vijay Kumar. Minimum snap trajectory generation and control for quadrotors. In *2011 IEEE International Conference on Robotics and Automation*, pages 2520–2525. IEEE, May 2011.

[26] Taeyoung Lee, Melvin Leok, and N.H. McClamroch. Control of Complex Maneuvers for a Quadrotor UAV using Geometric Methods on SE(3). *Asian Journal of Control*, 2011.

[27] "Gumstix, Inc.," http://www.gumstix.com.

[28] Andrew I. Comport, Eric Marchand, Muriel Pressigout, and François Chaumette. Real-time markerless tracking for augmented reality: the virtual visual servoing framework. *IEEE transactions on visualization and computer graphics*, 12(4):615–28.

APPENDIX

A. Attitude Controller

The proof of the attitude controller is outlined below.

Proof: Follows from [26, Prop. 1], with the Lyapunov function

$$\mathcal{V}_R = \frac{D_3}{2} e_\Omega \cdot e_\Omega + K_R \Psi(R, R_d) + c_2 e_R \cdot e_\Omega, \quad (8)$$

with c_2 being a positive scalar, such that,

$$\mathbf{z}_\theta^T M_\theta \mathbf{z}_\theta \leq \mathcal{V}_R \leq \mathbf{z}_\theta^T M_\Theta \mathbf{z}_\theta, \quad (9)$$

$$\dot{\mathcal{V}}_R \leq -\mathbf{z}_\theta^T W_\theta \mathbf{z}_\theta, \quad (10)$$

where $\mathbf{z}_\theta = [\|e_R\|, \|e_\Omega\|]^T$, and M_θ, M_Θ, and W_θ are positive definite. ∎

B. Position Control

To demonstrate the stability of the position controller, we first set up the proof. Define $K_p', K_d', B, \alpha \in \mathbb{R}$ as,

$$K_p' = m\|J\| \left\|J^{-1}\right\| K_p \quad (11)$$

$$K_d' = \|J\| \left(m \left\|J^{-1}\right\| K_d + \left\|\dot{J^{-1}}\right\|\right) \quad (12)$$

$$B = \|J\| \left(\|G_A\| + m \left\|J^{-1}\right\| \left\|\ddot{\mathbf{v}}^d\right\| + \left\|\dot{J^{-1}}\right\| \left\|\dot{\mathbf{v}}^d\right\|\right) \quad (13)$$

$$\alpha = \|e_R\| \quad (14)$$

and define $W_{v_1}, W_{v_2}, W_{v\theta}, W_v \in \mathbb{R}^{2\times 2}$ as

$$W_{v_1} = \begin{bmatrix} \frac{c_1 K_p}{m} & \frac{c_1 K_d}{2m} \\ \frac{c_1 K_d}{2m} & K_d - c_1 \end{bmatrix}, W_{v\theta} = \begin{bmatrix} \frac{c_1}{m} B & 0 \\ B & 0 \end{bmatrix} \quad (15)$$

$$W_{v_2} = \begin{bmatrix} \frac{c_1 \alpha K_p'}{m} & \frac{\alpha}{2}\left(\frac{c_1}{m} K_d' + K_p'\right) \\ \frac{\alpha}{2}\left(\frac{c_1}{m} K_d' + K_p'\right) & \alpha K_d' \end{bmatrix} \quad (16)$$

$$W_v = W_{v_1} - W_{v_2}. \quad (17)$$

Suppose we choose positive constants $c_1, K_p, K_d, K_R, K_\Omega$ such that,

$$K_p > \frac{c_1^2}{m} \quad (18)$$

$$\lambda_{min}(W_\theta) > \frac{4\|W_{v\theta}\|^2}{\lambda_{min}(W_v)} \quad (19)$$

Then, there exists positive constants $\gamma_1, \gamma_2, \gamma_3$, such that $\|J\| \leq \gamma_1, \left\|J^{-1}\right\| \leq \gamma_2, \left\|\dot{J^{-1}}\right\| \leq \gamma_3$, and if initial conditions and the desired trajectory satisfy

$$\alpha < \frac{1}{m\gamma_1\gamma_2}, \quad (20)$$

$$dist(\mathbf{v}_d(t), V^c) < \|\mathbf{e}_v(0)\|, \quad (21)$$

where V^c is the complement of V, and $dist(\mathbf{v}_d(t), V^c) = \inf_{t\in[0,\infty), w\in V^c} \|\mathbf{v}_d(t) - w\|$ is the smallest distance between a trajectory and a set, then the zero equilibrium $(\mathbf{e}_v, \dot{\mathbf{e}}_v, e_R, e_\Omega) = (\mathbf{0}, \mathbf{0}, 0, 0)$ is locally exponentially stable.

Proof: We take an approach very similar to [26] to show that the controller is exponentially stable. Using (5), we can determine the image errors

$$\ddot{\mathbf{e}}_v = \ddot{\mathbf{v}} - \ddot{\mathbf{v}}_d \quad (22)$$

$$= \frac{1}{m} J[f R\mathbf{e}_2 - \mathbf{G}_A] + \dot{J}J^{-1}\dot{\mathbf{v}} - \ddot{\mathbf{v}}_d \quad (23)$$

so that

$$m\ddot{\mathbf{e}}_v = fJR\mathbf{e}_2 - J\mathbf{G}_A + m\dot{J}J^{-1}\dot{\mathbf{v}} - m\ddot{\mathbf{v}}_d. \quad (24)$$

Defining

$$\mathbf{X} = J\frac{f}{\mathbf{e}_2^T R_c^T R\mathbf{e}_2}\left(\left(\mathbf{e}_2^T R_c^T R\mathbf{e}_2\right) R\mathbf{e}_2 - R_c\mathbf{e}_2\right), \quad (25)$$

the error dynamics become

$$m\ddot{\mathbf{e}}_v = J\left(\frac{f}{\mathbf{e}_2^T R_c^T R\mathbf{e}_2} R_c\mathbf{e}_2\right) + \mathbf{X} - J\mathbf{G}_A + m\dot{J}J^{-1}\dot{\mathbf{v}} - m\ddot{\mathbf{v}}_d. \quad (26)$$

Next, let

$$f = \mathbf{A}\cdot R\mathbf{e}_2 \quad (27)$$

and the commanded attitude be defined by

$$R_c\mathbf{e}_2 = \frac{\mathbf{A}}{\|\mathbf{A}\|}. \quad (28)$$

Then, from the previous two equations, we have

$$f = \|\mathbf{A}\| \mathbf{e}_2^T R_c^T R\mathbf{e}_2. \quad (29)$$

Substituting this into (26) and using $\mathbf{A}$, we have

$$m\ddot{\mathbf{e}}_v = J\left(\frac{\|\mathbf{A}\| \mathbf{e}_2^T R_c^T R\mathbf{e}_2}{\mathbf{e}_2^T R_c^T R\mathbf{e}_2} R_c\mathbf{e}_2\right) + \mathbf{X} - J\mathbf{G}_A + m\dot{J}J^{-1}\dot{\mathbf{v}} - m\ddot{\mathbf{v}}_d \quad (30)$$

$$= J(\|\mathbf{A}\| R_c\mathbf{e}_2) + \mathbf{X} - J\mathbf{G}_A + m\dot{J}J^{-1}\dot{\mathbf{v}} - m\ddot{\mathbf{v}}_d \quad (31)$$

$$= J\mathbf{A} + \mathbf{X} - J\mathbf{G}_A + m\dot{J}J^{-1}\dot{\mathbf{v}} - m\ddot{\mathbf{v}}_d \quad (32)$$

$$= -K_p\mathbf{e}_v - K_d\dot{\mathbf{e}}_v + \mathbf{X} \quad (33)$$

which has the same form as (83) in [26]. We use the same Lyapunov candidate, but in our image coordinates,

$$\mathcal{V}_v = \frac{1}{2}K_p\|\mathbf{e}_v\|^2 + \frac{1}{2}m\|\dot{\mathbf{e}}_v\|^2 + c_1\mathbf{e}_v\cdot\dot{\mathbf{e}}_v. \quad (34)$$

Now, let $\mathbf{z}_v = \left[\ \|\mathbf{e}_v\|, \|\dot{\mathbf{e}}_v\|\ \right]^T$, then it follows that the Lyapunov function $\mathcal{V}_v$ is bounded as

$$\mathbf{z}_v^T M_v \mathbf{z}_v \leq \mathcal{V}_v \leq \mathbf{z}_v^T M_V \mathbf{z}_v, \quad (35)$$

where $M_v, M_V \in \mathbb{R}^{2\times 2}$ are defined as,

$$M_v = \frac{1}{2}\begin{bmatrix} K_p & -c_1 \\ -c_1 & m \end{bmatrix}, \ M_V = \frac{1}{2}\begin{bmatrix} K_p & c_1 \\ c_1 & m \end{bmatrix}. \quad (36)$$

Then,

$$\dot{\mathcal{V}}_v = K_p(\dot{\mathbf{e}}_v\cdot\mathbf{e}_v) + m(\ddot{\mathbf{e}}_v\cdot\dot{\mathbf{e}}_v) + c_1(\mathbf{e}_v\cdot\ddot{\mathbf{e}}_v + \dot{\mathbf{e}}_v\cdot\dot{\mathbf{e}}_v), \quad (37)$$

and incorporating (33),

$$\dot{\mathcal{V}}_v = -\frac{c_1 K_p}{m}\|\mathbf{e}_v\|^2 - (K_d - c_1)\|\dot{\mathbf{e}}_v\|^2 - c_1\frac{K_d}{m}(\mathbf{e}_v\cdot\dot{\mathbf{e}}_v) + \mathbf{X}\cdot\left(\tfrac{c_1}{m}\mathbf{e}_v + \dot{\mathbf{e}}_v\right). \quad (38)$$

Now, we establish a bound on $\mathbf{X}$. From (25),

$$\mathbf{X} = J\frac{f}{\mathbf{e}_2^T R_c^T R\mathbf{e}_2}\left(\left(\mathbf{e}_2^T R_c^T R\mathbf{e}_2\right) R\mathbf{e}_2 - R_c\mathbf{e}_2\right) \tag{39}$$

$$\|\mathbf{X}\| \leq \|J\| \left\| \frac{\|A\| R_c\mathbf{e}_2 \cdot R\mathbf{e}_2}{R_c\mathbf{e}_2 \cdot R\mathbf{e}_2} \right\| \|e_R\| \tag{40}$$

$$\leq \|J\| \|\mathbf{A}\| \|e_R\| \tag{41}$$

$$\leq \|J\| \left\| \mathbf{G}_A + mJ^{-1}\left[-K_p\mathbf{e}_v - K_d\dot{\mathbf{e}}_v + \ddot{\mathbf{v}}_d\right] + \dot{J^{-1}}\left[\dot{\mathbf{e}}_v + \dot{\mathbf{v}}_d\right] \right\| \|e_R\| \tag{42}$$

$$\leq \left(K_p' \|\mathbf{e}_v\| + K_d' \|\dot{\mathbf{e}}_v\| + B\right) \|e_R\| \tag{43}$$

where K_p', K_d', B are as defined in (11)-(13), and from [26], $0 \leq \|e_R\| \leq 1$.

Next we will show that there exists positive constants $\gamma_1, \gamma_2, \gamma_3$ s.t., $\|J\| \leq \gamma_1, \left\|J^{-1}\right\| \leq \gamma_2$, and $\left\|\dot{J^{-1}}\right\| \leq \gamma_3$. Since Γ is smooth (we only require C^2 here), J is smooth on the closed set S. This implies J is bounded on S, i.e., $\exists \gamma_1 > 0$, s.t. $\|J\| < \gamma_1$. Next, since J is smooth and nonsingular on S, the inverse is well defined and is smooth on S, which implies J^{-1} is bounded on S, i.e., $\exists \gamma_2 > 0$, s.t. $\left\|J^{-1}\right\| < \gamma_2$. Next, observe that $\frac{d}{dt}J^{-1}(\mathbf{r}_q) = \frac{\partial}{\partial \mathbf{r}_q}J^{-1}(\mathbf{r}_q)\dot{\mathbf{r}}_q$ is a composition of smooth functions on S, implying that it is bounded on S, i.e., $\exists \gamma_3 > 0$, s.t. $\left\|\dot{J^{-1}}\right\| < \gamma_3$.

Then, similar to [10], we can express $\dot{\mathcal{V}}_v$ as

$$\dot{\mathcal{V}}_v = -\begin{bmatrix} \mathbf{e}_v^T & \dot{\mathbf{e}}_v^T \end{bmatrix} W_{v_1} \begin{bmatrix} \mathbf{e}_v \\ \dot{\mathbf{e}}_v \end{bmatrix} + \mathbf{X} \cdot \left(\frac{c_1}{m}\mathbf{e}_v + \dot{\mathbf{e}}_v\right) \tag{44}$$

$$\leq -\begin{bmatrix} \mathbf{e}_v^T & \dot{\mathbf{e}}_v^T \end{bmatrix} W_{v_1} \begin{bmatrix} \mathbf{e}_v \\ \dot{\mathbf{e}}_v \end{bmatrix} + K_p' \|\mathbf{e}_v\| \|e_R\| \left(\tfrac{c_1}{m}\|\mathbf{e}_v\| + \|\dot{\mathbf{e}}_v\|\right) + K_d' \|\dot{\mathbf{e}}_v\| \|e_R\| \left(\tfrac{c_1}{m}\|\mathbf{e}_v\| + \|\dot{\mathbf{e}}_v\|\right) + B\|e_R\| \left(\tfrac{c_1}{m}\|\mathbf{e}_v\| + \|\dot{\mathbf{e}}_v\|\right). \tag{45}$$

This can be written as,

$$\dot{\mathcal{V}}_v \leq -\mathbf{z}_v^T W_v \mathbf{z}_v + \mathbf{z}_v^T W_{v\theta} \mathbf{z}_\theta \tag{46}$$

where $W_{v\theta}, W_v$ are as defined in (15), (17). Since $W_v = (W_v)^T$ and $W_v \in \mathbb{R}^{2\times 2}$, it is sufficient to show that $\det(W_v) > 0$ and $W_v(1,1) > 0$ in order to claim that $W_v > 0$. Then, from the assumption on α in (20), we have $w_{11} > 0$. This is reasonable since α is a functional on the attitude error such that $\alpha \in [0, 1]$. Thus, the assumption in (20) is simply a bound on the attitude error. The determinant can be expressed as a quadratic function of K_d such that

$$\det(W_v) = \beta_0 + \beta_1 K_d + \beta_2 K_d^2 \tag{47}$$

and β_i is a function of c_1, K_p, γ_1, γ_2, γ_3, and m. The critical point of the quadratic occurs when

$$K_d = \frac{K_p m}{c_1} + \frac{K_p m + \alpha c_1 \gamma_1 \gamma_3}{c_1\left(1 - \alpha\gamma_1\gamma_2 m\right)} \tag{48}$$

and has a value of

$$\det(W_v) = \frac{K_p\left(1 - \alpha\gamma_1\gamma_2 m\right)\left(K_p m - c_1^2\right)}{m}. \tag{49}$$

In both equations, $(1 - \alpha\gamma_1\gamma_2 m) > 0$ as a result of the assumption in (20). Thus (48) is positive, and by (18), (49) is positive and $W_v' > 0$. Now, we consider the combined Lyapunov candidate for the translational and rotational error dynamics, $\mathcal{V} = \mathcal{V}_v + \mathcal{V}_R$. From (9) and (35), we have,

$$\mathbf{z}_v^T M_v \mathbf{z}_v + \mathbf{z}_\theta^T M_\theta \mathbf{z}_\theta \leq \mathcal{V} \leq \mathbf{z}_\theta^T M_\Theta \mathbf{z}_\theta + \mathbf{z}_v^T M_V \mathbf{z}_v. \tag{50}$$

Further, we see that

$$\dot{\mathcal{V}} \leq -\mathbf{z}_v^T W_v \mathbf{z}_v + \mathbf{z}_v^T W_{v\theta}\mathbf{z}_\theta - \mathbf{z}_\theta^T W_\theta \mathbf{z}_\theta, \tag{51}$$

$$\leq -\lambda_{min}(W_v)\|\mathbf{z}_v\|^2 + \|W_{v\theta}\|\|\mathbf{z}_v\|\|\mathbf{z}_\theta\| - \lambda_{min}(W_\theta)\|\mathbf{z}_\theta\|^2, \tag{52}$$

and from (19), we have $\dot{\mathcal{V}}$ to be negative definite, and the zero equilibrium of the closed-loop system is locally exponentially stable. ■