2005 American Control Conference
June 8-10, 2005. Portland, OR, USA

Navigation Function Based Visual Servo Control[1]

J. Chen,[†] D. M. Dawson[†], W. E. Dixon,[‡] and V. K. Chitrakaran[†]

[†]Department of Electrical & Computer Engineering, Clemson University, Clemson, SC 29634-0915

[‡]Department of Mechanical & Aerospace Engineering, University of Florida, Gainesville, FL 32611

email: jian.chen@ieee.org, ddawson@ces.clemson.edu, wdixon@ufl.edu, cvilas@ces.clemson.edu

Abstract: *In this paper, the mapping between the desired camera feature vector and the desired camera pose (i.e., the position and orientation) is investigated to develop a measurable image Jacobian-like matrix. An image-space path planner is then proposed to generate a desired image trajectory based on this measurable image Jacobian-like matrix and an image space navigation function (NF) (i.e., a special potential field function) while satisfying rigid body constraints. An adaptive, homography-based visual servo tracking controller is then developed to navigate the position and orientation of a camera held by the end-effector of a robot manipulator to a goal position and orientation along the desired image-space trajectory while ensuring the target points remain visible (i.e., the target points avoid self-occlusion and remain in the field-of-view (FOV)) under certain technical restrictions. Due to the inherent nonlinear nature of the problem and the lack of depth information from a monocular system, a Lyapunov-based analysis is used to analyze the path planner and the adaptive controller.*

1 Introduction

There is significant motivation to provide improved autonomy for robotic systems. In part, this motivation has led researchers to investigate the basic science challenges leading to the development of visual servo controllers as a means to provide improved robot autonomy. In general, visual servo controllers can be divided into position-based visual servo (PBVS) control, image-based visual servo (IBVS), and hybrid approaches. PBVS is based on the idea of using a vision system to reconstruct the Euclidean-space and then developing the servo controller on the reconstructed information. A well known issue with this strategy is that the target object may exit the camera field-of-view (FOV). IBVS control is based on the idea of directly servoing on the image-space information, with reported advantages of increased robustness to camera calibration and improved capabilities to ensure the target remains visible. Even for IBVS controllers that are formulated as regulation controllers, if the initial error is large, then excessive control action and transient response can cause the target to leave the FOV, and may lead to trajectories that are not physically valid or optimal due to the nonlinearities and potential singularities with associated the transformation between the image space and the Euclidean-space [2]. For a review of IBVS and PBVS controllers see [20].

In light of the characteristics of IBVS and PBVS, several researchers have recently explored hybrid approaches. For example, homography-based visual servo control techniques (coined 2.5D controllers) have been recently developed in a series of papers by Malis and Chaumette (e.g., [1], [26], [27]). The homography-based approach exploits a combination of reconstructed Euclidean information and image-space information in the control design. The Euclidean information is reconstructed by decoupling the interaction between translational and rotational components of a homography matrix. As stated in [26], some advantages of this methodology over the aforementioned IBVS and PBVS approaches are that an accurate Euclidean model of the environment (or target object) is not required, and potential singularities in the image-Jacobian are eliminated (i.e., the image-Jacobian for homography-based visual servo controllers is typically triangular). Motivated by the advantages of the homography-based strategy, several researchers have recently developed various regulation controllers for robot manipulators (see [5], [8], and [11]).

While homography-based approaches exploit the advantages of IBVS and PBVS, a common problem with all the aforementioned approaches is the inability to achieve the control objective while ensuring the target features remain visible. To address this issue, Mezouar and Chaumette developed a path-following IBVS algorithm in [29] where the path to a goal point is generated via a potential function that incorporates motion constraints; however, as stated in [29], local minima associated with traditional potential functions may exist. Using a specialized potential function (coined a navigation function (NF)) originally proposed in [24] and

[1]This work is supported in part by two DOC Grants, an ARO Automotive Center Grant, a DOE Contract, a Honda Corporation Grant, U.S. NSF Grant DMI-9457967, ONR Grant N00014-99-1-0589, and a DARPA Contract at Clemson University, and in part by AFOSR contract number F49620-03-1-0381 at the University of Florida.

[34], Cowan et al. developed a hybrid position/image-space controller that forces a manipulator to a desired setpoint while ensuring the object remains visible (i.e., the NF ensures no local minima) and by avoiding pitfalls such as self-occlusion [10]. However, as stated in [29], this approach requires the complete knowledge of the space topology and requires an object model. In [18], Gans and Hutchinson developed a strategy that switches between an IBVS and a PBVS controller to ensure asymptotic stability of the position and orientation (i.e., pose) in the Euclidean and image-space. An image-space based follow-the-leader application for mobile robots was developed in [9] that exploits an image-space NF. Specifically, an input/output feedback linearization technique is applied to the mobile robot kinematic model to yield a controller that yields "string stability" [16]. Without a feedforward component, the controller in [9] yields an approximate "input-to-formation" stability (i.e., a local, linear exponential system with a bounded disturbance). A NF based approach to the follow-the-leader problem for a group of fully actuated holonomic mobile robots is considered in [31] where configuration based constraints are developed to ensure the robot edges remain in the sight of an omnidirectional camera. While a Lyapunov-based analysis is provided in [31] to ensure that the NF decreases to the goal position, the stability of the overall system is not examined.

Motivated by the image space navigation function developed in [10], an off-line desired image trajectory generator is proposed based on a new image Jacobian-like matrix for the monocular, camera-in-hand problem. This approach generates a desired camera pose trajectory that moves the camera from the initial camera pose to a goal camera pose while ensuring that all the feature points of the object remain visible under certain technical restrictions. To develop a desired camera pose trajectory that ensures all feature points remain visible, a unique relationship is formulated between the desired image feature vector and the desired camera pose. The resulting image Jacobian-like matrix is related to the camera pose, rather than the camera velocity as in other approaches [2]. Motivation for the development of this relationship is that the resulting image Jacobian-like matrix is measurable, and hence, does not suffer from the lack of robustness associated with estimation based methods. Further more, the desired image generated with this image Jacobian-like matrix satisfies rigid body constraints (The terminology, rigid body constraints, in this paper is utilized to denote the image feature vector constraints in which feature points have a fixed relative position to each other in Euclidean space). Building on our recent research in [5], an adaptive homography based visual tracking controller is then developed to ensure that the actual camera pose tracks the desired camera pose trajectory (i.e., the actual features track the desired feature point trajectory) despite the fact that time-varying depth from the camera to the reference image plane is not measurable from the monocular camera system. Based on the analysis of the homography based controller, bounds are developed that can be used to ensure that the actual image features also remain visible under certain technical restrictions. A Lyapunov-based analysis is provided to support the claims for the path planner and to analyze the stability of the adaptive tracking controller.

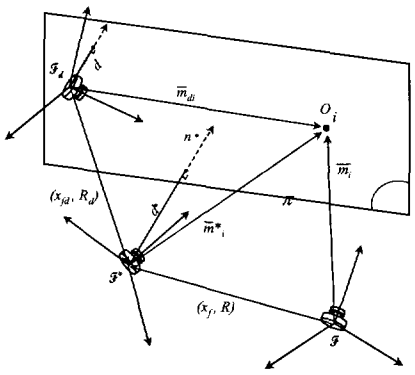

Figure 1: Coordinate frame relationships

2 Geometric Modeling

2.1 Euclidean Homography

Four feature points, denoted by O_i $\forall i = 1, 2, 3, 4$, are assumed to be located on a reference plane π (see Figure 1), and are considered to be coplanar[1] and not colinear. The reference plane can be related to the coordinate frames $\mathcal{F}$, $\mathcal{F}_d$, and $\mathcal{F}^*$ depicted in Fig. 1 that denote the actual, desired, and goal pose of the camera, respectively. Specifically, the following relationships can be developed from the geometry between the coordinate frames and the feature points located on π

$$\begin{aligned} \bar{m}_i &= x_f + R\bar{m}_i^* \\ \bar{m}_{di} &= x_{fd} + R_d\bar{m}_i^* \end{aligned} \tag{1}$$

where $\bar{m}_i(t)$, $\bar{m}_{di}(t)$, and $\bar{m}_i^*$ denote the Euclidean coordinates of O_i expressed in $\mathcal{F}$, $\mathcal{F}_d$, and $\mathcal{F}^*$, respectively. In (1), $R(t)$, $R_d(t) \in SO(3)$ denote the rotation between $\mathcal{F}$ and $\mathcal{F}^*$ and between $\mathcal{F}_d$ and $\mathcal{F}^*$, respectively, and $x_f(t)$, $x_{fd}(t) \in \mathbb{R}^3$ denote translation vectors from $\mathcal{F}$ to $\mathcal{F}^*$ and $\mathcal{F}_d$ to $\mathcal{F}^*$ expressed in the coordinates of $\mathcal{F}$ and $\mathcal{F}_d$, respectively. Since the Euclidean position

[1] It should be noted that if four coplanar target points are not available then the subsequent development can exploit the classic eight-points algorithm [27] with no four of the eight target points being coplanar.

of $\mathcal{F}$, $\mathcal{F}_d$, and $\mathcal{F}^*$ cannot be directly measured, the expressions in (1) need to be related to the measurable image-space coordinates. To this end, the normalized Euclidean coordinates of O_i expressed in terms of $\mathcal{F}$, $\mathcal{F}_d$, and $\mathcal{F}^*$ as $m_i(t)$, $m_{di}(t)$, $m_i^* \in \mathbb{R}^3$, respectively, are defined as follows

$$m_i \triangleq \frac{\bar{m}_i}{z_i} \qquad m_{di} \triangleq \frac{\bar{m}_{di}}{z_{di}} \qquad m_i^* \triangleq \frac{\bar{m}_i^*}{z_i^*} \tag{2}$$

under the standard assumption that $z_i(t)$, $z_{di}(t)$, $z_i^* > \varepsilon$ where ε denotes an arbitrarily small positive constant. Based on (2), the expression in (1) can be rewritten as follows

$$m_i = \underbrace{\frac{z_i^*}{z_i}}_{\alpha_i} \underbrace{\left(R + \frac{x_f}{d^*} n^{*T}\right)}_{H} m_i^* \tag{3}$$

$$m_{di} = \underbrace{\frac{z_i^*}{z_{di}}}_{\alpha_{di}} \underbrace{\left(R_d + \frac{x_{fd}}{d^*} n^{*T}\right)}_{H_d} m_i^*. \tag{4}$$

In (3) and (4), $\alpha_i(t)$, $\alpha_{di}(t) \in \mathbb{R}$ denote invertible depth ratios, $H(t)$, $H_d(t) \in \mathbb{R}^{3\times 3}$ denote Euclidean homographies [14], and $d^* \in \mathbb{R}$ denotes the constant, unknown distance from the origin of $\mathcal{F}^*$ to π. The following projective relationship can also be developed from Fig. 1

$$d^* = n^{*T} \bar{m}_i^*. \tag{5}$$

Also from Fig. 1, the unknown, time varying distance from the origin of $\mathcal{F}_d$ to π, denoted by $d(t) \in \mathbb{R}$, can be expressed as follows

$$d = n^{*T} R_d^T \bar{m}_{di}. \tag{6}$$

2.2 Projective Homography

Each feature point on π has a projected pixel coordinate denoted by $u_i(t)$, $v_i(t) \in \mathbb{R}$ in $\mathcal{F}$, $u_{di}(t)$, $v_{di}(t) \in \mathbb{R}$ in $\mathcal{F}_d$, and u_i^*, $v_i^* \in \mathbb{R}$ in $\mathcal{F}^*$, that are defined as follows

$$p_i \triangleq \begin{bmatrix} u_i & v_i & 1 \end{bmatrix}^T \qquad p_{di} \triangleq \begin{bmatrix} u_{di} & v_{di} & 1 \end{bmatrix}^T$$
$$p_i^* \triangleq \begin{bmatrix} u_i^* & v_i^* & 1 \end{bmatrix}^T. \tag{7}$$

In (7), $p_i(t)$, $p_{di}(t)$, $p_i^* \in \mathbb{R}^3$ represent the image-space coordinates of the time-varying feature points, the desired time-varying feature point trajectory, and the constant reference feature points, respectively. To calculate the Euclidean homography given in (3) and (4) from pixel information, the projected pixel coordinates of the target points are related to $m_i(t)$, $m_{di}(t)$, and m_i^* by the following pin-hole lens models [14]

$$p_i = A m_i \qquad p_{di} = A m_{di} \qquad p_i^* = A m_i^* \tag{8}$$

where $A \in \mathbb{R}^{3\times 3}$ is a known, constant, and invertible intrinsic camera calibration matrix. After substituting (8) into (3) and (4), the following relationships can be developed

$$p_i = \alpha_i \underbrace{\left(AHA^{-1}\right)}_{G} p_i^* \qquad p_{di} = \alpha_{di} \underbrace{\left(AH_dA^{-1}\right)}_{G_d} p_i^* \tag{9}$$

where $G(t)$, $G_d(t) \in \mathbb{R}^{3\times 3}$ denote projective homographies. Given the images of the 4 feature points on π expressed in $\mathcal{F}$, $\mathcal{F}_d$, and $\mathcal{F}^*$, a linear system of equations can be developed from (9). From the linear system of equations, a decomposition algorithm (e.g., the Faugeras decomposition algorithm in [14]) can be used to compute $\alpha_i(t)$, $\alpha_{di}(t)$, n^*, $R(t)$, and $R_d(t)$ (see [5] for details)[2]. Hence, $\alpha_i(t)$, $\alpha_{di}(t)$, n^*, $R(t)$, and $R_d(t)$ are known signals that can be used in the subsequent development.

2.3 Kinematic Model of Vision System

The camera pose, denoted by $\Upsilon(t) \in \mathbb{R}^6$, can be expressed in terms of a hybrid of pixel and reconstructed Euclidean information as follows

$$\Upsilon(t) \triangleq \begin{bmatrix} p_{e1}^T & \Theta^T \end{bmatrix}^T \tag{10}$$

where the extended pixel coordinate $p_{e1}(t) \in \mathbb{R}^3$ is defined as follows

$$p_{e1} = \begin{bmatrix} u_1 & v_1 & -\ln(\alpha_1) \end{bmatrix}^T, \tag{11}$$

and $\Theta(t) \in \mathbb{R}^3$ denotes the following axis-angle representation of $R(t)$ (see [5] for details)

$$\Theta = \mu(t)\theta(t). \tag{12}$$

In (11), $\ln(\cdot)$ denotes the natural logarithm, and $\alpha_1(t)$ is introduced in (3). In (12), $\mu(t) \in \mathbb{R}^3$ represents the unit axis of rotation, and $\theta(t)$ denotes the rotation angle about that axis. Based on the development in Appendix A, the open-loop dynamics for $\Upsilon(t)$ can be expressed as follows

$$\dot{\Upsilon} = \begin{bmatrix} \dot{p}_{e1} \\ \dot{\Theta} \end{bmatrix} = \begin{bmatrix} -\frac{1}{z_1} A_{e1} & A_{e1}[m_1]_\times \\ 0 & -L_\omega \end{bmatrix} \begin{bmatrix} v_c \\ \omega_c \end{bmatrix} \tag{13}$$

where $v_c(t) \in \mathbb{R}^3$ and $\omega_c(t) \in \mathbb{R}^3$ denote the linear and angular velocity of the camera expressed in terms of $\mathcal{F}$, $A_{ei}(u_i, v_i) \in \mathbb{R}^{3\times 3}$ is a known, invertible matrix defined as follows

$$A_{ei} = A - \begin{bmatrix} 0 & 0 & u_i \\ 0 & 0 & v_i \\ 0 & 0 & 0 \end{bmatrix} \qquad i = 1, 2, 3, 4, \tag{14}$$

and the invertible Jacobian-like matrix $L_\omega(\theta, \mu) \in \mathbb{R}^{3\times 3}$ is defined as [26]

$$L_\omega = I_3 - \frac{\theta}{2}[\mu]_\times + \left(1 - \frac{sinc(\theta)}{sinc^2\left(\frac{\theta}{2}\right)}\right)[\mu]_\times^2 \tag{15}$$

[2] The initial best-guess of n^* can be utilized to resolve the decomposition ambiguity. See [7] for details.

where

$$sinc(\theta(t)) \triangleq \frac{\sin\theta(t)}{\theta(t)}.$$

3 Image-Based Path Planning

The path planning objective involves regulating the pose of a camera held by the end-effector of a robot manipulator to a desired camera pose along an image-space trajectory while ensuring the target points remain visible. To achieve this objective, a desired camera pose trajectory is constructed in this section so that the desired image feature vector, denoted by $\bar{p}_d(t) \triangleq \left[\begin{array}{cccccc} u_{d1}(t) & v_{d1}(t) & \cdots & u_{d4}(t) & v_{d4}(t) \end{array}\right]^T \in \mathbb{R}^8$, remains in a set, denoted by $\mathcal{D} \subset \mathbb{R}^8$, where all four feature points of the target remain visible for a valid camera pose. The constant, goal image feature vector $\bar{p}^* \triangleq \left[\begin{array}{cccccc} u_1^* & v_1^* & \cdots & u_4^* & v_4^* \end{array}\right]^T \in \mathbb{R}^8$ is assumed be in the interior of $\mathcal{D}$. To generate the desired camera pose trajectory such that $\bar{p}_d(t) \in \mathcal{D}$, the special artificial potential function coined a navigation function in [24], can be used. Specifically, the navigation functions used in this paper are defined as follows [34].

Definition 1 *Let $\mathcal{D}$ be a compact connected analytic manifold with boundary, and let $\bar{p}^*$ be a goal point in the interior of $\mathcal{D}$. A map $\varphi(\bar{p}_d) : \mathcal{D} \rightarrow [0, 1]$, is a NF if it is*

P 1) analytic on $\mathcal{D}$ (at least the first and second partial derivatives exist and are bounded on $\mathcal{D}$);

P 2) polar at $\bar{p}^*$, i.e., has a unique minimum at $\bar{p}^*$;

P 3) admissible on $\mathcal{D}$, i.e., uniformly maximal on the boundary of $\mathcal{D}$;

P 4) a Morse function (i.e., the matrix of second partial derivatives, the Hessian, evaluated at its critical points is nonsingular (and has bounded elements based on the smoothness property in P 1)).

3.1 Pose Space to Image Space Relationship

To develop a desired camera pose trajectory that ensures $\bar{p}_d(t) \in \mathcal{D}$, the desired image feature vector is related to the desired camera pose, denoted by $\Upsilon_d(t) \in \mathbb{R}^6$, through the following relationship

$$\bar{p}_d = \Pi(\Upsilon_d) \tag{16}$$

where $\Pi(\cdot) : \mathbb{R}^6 \rightarrow \mathcal{D}$ denotes an unknown function that maps the camera pose to the image feature vector[3]. In (16), the desired camera pose is defined as follows

$$\Upsilon_d(t) \triangleq \left[\begin{array}{cc} p_{ed1}^T & \Theta_d^T \end{array}\right]^T \tag{17}$$

where $p_{ed1}(t) \in \mathbb{R}^3$ denotes the desired extended pixel coordinates defined as follows

$$p_{ed1} = \left[\begin{array}{ccc} u_{d1} & v_{d1} & -\ln(\alpha_{d1}) \end{array}\right]^T \tag{18}$$

where $\alpha_{d1}(t)$ is introduced in (4), and $\Theta_d(t) \in \mathbb{R}^3$ denotes the axis-angle representation of $R_d(t)$ as follows

$$\Theta_d = \mu_d(t)\theta_d(t) \tag{19}$$

where $\mu_d(t) \in \mathbb{R}^3$ and $\theta_d(t) \in \mathbb{R}$ are defined in the same manner as $\mu(t)$ and $\theta(t)$ in (12) with respect to $R_d(t)$.

3.2 Desired Image Trajectory Planning

After taking the time derivative of (16), the following expression can be obtained

$$\dot{\bar{p}}_d = L_{\Upsilon_d}\dot{\Upsilon}_d \tag{20}$$

where $L_{\Upsilon_d}(\bar{p}_d) \triangleq \dfrac{\partial \bar{p}_d}{\partial \Upsilon_d} \in \mathbb{R}^{8\times 6}$ denotes an image Jacobian-like matrix. Based on the development in Appendix B, a measurable expression for $L_{\Upsilon_d}(t)$ can be developed as follows

$$L_{\Upsilon_d} = \bar{I}T \tag{21}$$

where $\bar{I} \in \mathbb{R}^{8\times 12}$ denotes a constant, row-delete matrix defined as follows

$$\bar{I} = \left[\begin{array}{cccccccc} I_2 & 0^2 & 0_2 & 0^2 & 0_2 & 0^2 & 0_2 & 0^2 \\ 0_2 & 0^2 & I_2 & 0^2 & 0_2 & 0^2 & 0_2 & 0^2 \\ 0_2 & 0^2 & 0_2 & 0^2 & I_2 & 0^2 & 0_2 & 0^2 \\ 0_2 & 0^2 & 0_2 & 0^2 & 0_2 & 0^2 & I_2 & 0^2 \end{array}\right]$$

where $I_n \in \mathbb{R}^{n\times n}$ denotes the $n \times n$ identity matrix, $0_n \in \mathbb{R}^{n\times n}$ denotes an $n \times n$ matrix of zeros, $0^n \in \mathbb{R}^n$ denotes an $n \times 1$ column of zeros, and $T(t) \in \mathbb{R}^{12\times 6}$ is a measurable auxiliary matrix defined as follows

$$T = \left[\begin{array}{cc} I_3 & 0_3 \\ \frac{\beta_1}{\beta_2}A_{ed2}A_{ed1}^{-1} & A_{ed2}\left[\frac{\beta_1}{\beta_2}m_{d1} - m_{d2}\right]_\times L_{\omega d}^{-1} \\ \frac{\beta_1}{\beta_3}A_{ed3}A_{ed1}^{-1} & A_{ed3}\left[\frac{\beta_1}{\beta_3}m_{d1} - m_{d3}\right]_\times L_{\omega d}^{-1} \\ \frac{\beta_1}{\beta_4}A_{ed4}A_{ed1}^{-1} & A_{ed4}\left[\frac{\beta_1}{\beta_4}m_{d1} - m_{d4}\right]_\times L_{\omega d}^{-1} \end{array}\right]. \tag{22}$$

In (22), $A_{edi}(u_{di}, v_{di}) \in \mathbb{R}^{3\times 3}$ and the Jacobian-like matrix $L_{\omega d}(\theta_d, \mu_d) \in \mathbb{R}^{3\times 3}$ are defined as in (14) and (15) with respect to $u_{di}(t), v_{di}(t), \mu_d(t)$, and $\theta_d(t)$, respectively. The auxiliary variable $\beta_i(t) \in \mathbb{R}$ in (22) is defined as follows

$$\beta_i \triangleq \frac{z_{di}}{d} \quad i = 1, 2, 3, 4. \tag{23}$$

[3] The reason we choose four feature points to construct the image feature vector is that the same image of three points can be seen from four different camera poses [21]. A unique camera pose can theoretically be obtained by using at least four points [2]. Therefore, the map $\Pi(\cdot)$ is a unique mapping with the image feature vector corresponding to a valid camera pose.

Based on (2), (6), and (8), $\beta_i(t)$ can be rewritten in terms of computed and measurable terms as follows

$$\beta_i = \frac{1}{n^{*T} R_d^T A^{-1} p_{di}}. \quad (24)$$

Motivated by (20) and the definition of the navigation function in Definition 1, the desired camera pose trajectory is designed as follows

$$\dot{\Upsilon}_d = -k_1 L_{\Upsilon_d}^T \bigtriangledown \varphi \quad (25)$$

where $k_1 \in \mathbb{R}$ denotes a positive constant, and $\bigtriangledown\varphi(\bar{p}_d) \triangleq \left(\frac{\partial \varphi(\bar{p}_d)}{\partial \bar{p}_d}\right)^T \in \mathbb{R}^8$ denotes the gradient vector of $\varphi(\bar{p}_d)$. The development of a particular image space NF and its gradient are provided in Appendix C. After substituting (25) into (20), the desired image trajectory can be expressed as follows

$$\dot{\bar{p}}_d = -k_1 L_{\Upsilon_d} L_{\Upsilon_d}^T \bigtriangledown \varphi \quad (26)$$

where it is assumed that $\bigtriangledown\varphi(\bar{p}_d)$ is not a member of the null space of $L_{\Upsilon_d}^T(\bar{p}_d)$. Based on (20) and (25), it is clear that the desired image trajectory generated by (26) will satisfy rigid body constraints.

Remark 1 *Based on comments in [2] and the current development, it seems that a remaining open problem is to develop a rigorous, theoretical and general approach to ensure that $\bigtriangledown\varphi(\bar{p}_d)$ is not a member of the null space of $L_{\Upsilon_d}^T(\bar{p}_d)$ (i.e., $\bigtriangledown\varphi(\bar{p}_d) \notin NS(L_{\Upsilon_d}^T(\bar{p}_d))$ where $NS(\cdot)$ denotes the null space operator). However, since the approach in this paper is developed in terms of the desired image-space trajectory (and hence, is an off-line approach), a particular desired image trajectory can be chosen (e.g., by trial and error) a priori to ensure that $\bigtriangledown\varphi(\bar{p}_d) \notin NS(L_{\Upsilon_d}^T(\bar{p}_d))$. Similar comments are provided in [2] and [29] that indicate that in practice this assumption can be readily satisfied for particular cases. Likewise, a particular desired image trajectory is also assumed to be a priori selected to ensure that $\Upsilon_d(t)$, $\dot{\Upsilon}_d(t) \in \mathcal{L}_\infty$ if $\bar{p}_d(t) \in \mathcal{D}$. Based on the structure of (17) and (18), the assumption that $\Upsilon_d(t)$, $\dot{\Upsilon}_d(t) \in \mathcal{L}_\infty$ if $\bar{p}_d(t) \in \mathcal{D}$ is considered mild in the sense that the only possible alternative case is if the camera could somehow be positioned at an infinite distance from the target while all four feature points remain visible.*

3.3 Path Planner Analysis

Theorem 1 *Provided the desired feature points can be a priori selected to ensure that $\bar{p}_d(0) \in \mathcal{D}$ and that $\bigtriangledown\varphi(\bar{p}_d) \notin NS(L_{\Upsilon_d}^T(\bar{p}_d))$, then the desired image trajectory generated by (26) ensures that $\bar{p}_d(t) \in \mathcal{D}$ and (26) has the asymptotically stable equilibrium point $\bar{p}^*$.*

Proof: See [6].

4 Tracking Control Development

Based on Theorem 1, the desired camera pose trajectory can be generated from (25) to ensure that the camera moves along a path generated in the image space such that the desired object features remain visible (i.e., $\bar{p}_d(t) \in \mathcal{D}$). The objective in this section is to develop a controller so that the actual camera pose $\Upsilon(t)$ tracks the desired camera pose $\Upsilon_d(t)$ generated by (25), while also ensuring that the object features remain visible (i.e., $\bar{p}(t) \triangleq \begin{bmatrix} u_1(t) & v_1(t) & \cdots & u_4(t) & v_4(t) \end{bmatrix}^T \in \mathcal{D}$). To quantify this objective, a rotational tracking error, denoted by $e_\omega(t) \in \mathbb{R}^3$, is defined as

$$e_\omega \triangleq \Theta - \Theta_d, \quad (27)$$

and a translational tracking error, denoted by $e_v(t) \in \mathbb{R}^3$, is defined as follows

$$e_v = p_{e1} - p_{ed1}. \quad (28)$$

4.1 Control Development

After taking the time derivative of (27) and (28), the open-loop dynamics for $e_\omega(t)$ and $e_v(t)$ can be obtained as follows

$$\dot{e}_\omega = -L_\omega \omega_c - \dot{\Theta}_d \quad (29)$$

$$\dot{e}_v = -\frac{1}{z_1} A_{e1} v_c + A_{e1} [m_1]_\times \omega_c - \dot{p}_{ed1} \quad (30)$$

where (13) was utilized. Based on the open-loop error systems in (29) and (30), $v_c(t)$ and $\omega_c(t)$ are designed as follows

$$\omega_c \triangleq L_\omega^{-1} \left(K_\omega e_\omega - \dot{\Theta}_d \right) \quad (31)$$

$$v_c \triangleq \frac{1}{\alpha_1} A_{e1}^{-1} \left(K_v e_v - \hat{z}_1^* \dot{p}_{ed1} \right) + \frac{1}{\alpha_1} [m_1]_\times \omega_c \hat{z}_1^* \quad (32)$$

where K_ω, $K_v \in \mathbb{R}^{3\times 3}$ denote diagonal matrices of positive constant control gains, and $\hat{z}_1^*(t) \in \mathbb{R}$ denotes a parameter estimate for z_1^* that is designed as follows

$$\dot{\hat{z}}_1^* \triangleq k_2 e_v^T \left(A_{e1} [m_1]_\times \omega_c - \dot{p}_{ed1} \right) \quad (33)$$

where $k_2 \in \mathbb{R}$ denotes a positive constant adaptation gain. After substituting (31) and (32) into (29) and (30), the following closed-loop error systems can be developed

$$\dot{e}_\omega = -K_\omega e_\omega \quad (34)$$

$$z_1^* \dot{e}_v = -K_v e_v + \left(A_{e1} [m_1]_\times \omega_c - \dot{p}_{ed1} \right) \tilde{z}_1^* \quad (35)$$

where the parameter estimation error signal $\tilde{z}_1^*(t) \in \mathbb{R}$ is defined as follows

$$\tilde{z}_1^* = z_1^* - \hat{z}_1^*. \quad (36)$$

4.2 Controller Analysis

Theorem 2 *The controller introduced in (31) and (32), along with the adaptive update law defined in (33), ensure that the actual camera pose tracks the desired camera pose trajectory in the sense that*

$$\|e_\omega(t)\| \to 0 \quad \|e_v(t)\| \to 0 \text{ as } t \to \infty. \tag{37}$$

Proof: Let $V_2(t) \in \mathbb{R}$ denote a non-negative function defined as follows

$$V_2 \triangleq \frac{1}{2} e_\omega^T e_\omega + \frac{z_1^*}{2} e_v^T e_v + \frac{1}{2k_2} \tilde{z}_1^{*2}. \tag{38}$$

After taking the time derivative of (38) and then substituting for the closed-loop error systems developed in (34) and (35), the following expression can be obtained

$$\begin{aligned} \dot{V}_2 = & -e_\omega^T K_\omega e_\omega - e_v^T K_v e_v \\ & + e_v^T \left(A_{e1} [m_1]_\times \omega_c - \dot{p}_{ed1} \right) \tilde{z}_1^* - \tfrac{1}{k_2} \tilde{z}_1^* \dot{\hat{z}}_1^* \end{aligned} \tag{39}$$

where the time derivative of (36) was utilized. After substituting the adaptive update law designed in (33) into (39), the following expression can be obtained

$$\dot{V}_2 = -e_\omega^T K_\omega e_\omega - e_v^T K_v e_v. \tag{40}$$

Standard signal chasing can be utilized to prove the result given in (37) (See [6] for the details). □

Remark 2 *Based on the result provided in (37), it can be proven from the Euclidean reconstruction given in (3) and (4) that $R(t) \to R_d(t)$, $m_1(t) \to m_{d1}(t)$, and $z_1(t) \to z_{d1}(t)$ (and hence, $x_f(t) \to x_{fd}(t)$). Based on these results, (1) can be used to also prove that $\bar{m}_i(t) \to \bar{m}_{di}(t)$. Since $\Pi(\cdot)$ is a unique mapping, we can conclude that the desired camera pose converges to the goal camera pose based on the previous result $\bar{p}_d(t) \to \bar{p}^*$ from Theorem 1. Based on the above analysis, $\bar{m}_i(t) \to \bar{m}^*$.*

Remark 3 *Based on (38) and (40), the following inequality can be obtained*

$$\begin{aligned} e_\omega^T e_\omega + e_v^T e_v &\leqslant 2 \max \left\{ 1, \frac{1}{z_1^*} \right\} V_2(t) \\ &\leqslant 2 \max \left\{ 1, \frac{1}{z_1^*} \right\} V_2(0) \end{aligned} \tag{41}$$

where

$$V_2(0) = \frac{1}{2} e_\omega^T(0) e_\omega(0) + \frac{z_1^*}{2} e_v^T(0) e_v(0) + \frac{1}{2k_2} \tilde{z}_1^{*2}(0).$$

From (10), (17), (27), (28), and the inequality in (41), the following inequality can be developed

$$\|\Upsilon - \Upsilon_d\| \leqslant \sqrt{2 \max \left\{ 1, \frac{1}{z_1^*} \right\} V_2(0)}. \tag{42}$$

Based on (16), the following expression can be developed

$$\bar{p} = \Pi(\Upsilon) - \Pi(\Upsilon_d) + \bar{p}_d. \tag{43}$$

After applying the mean-value theorem to (43), the following inequality can be obtained

$$\|\bar{p}\| \leqslant \|L_{\Upsilon_d}\| \|\Upsilon - \Upsilon_d\| + \|\bar{p}_d\|. \tag{44}$$

Since all signals are bounded, it can be shown that $L_{\Upsilon_d}^T(\bar{p}_d) \in \mathcal{L}_\infty$; hence, the following inequality can be developed from (42) and (44)

$$\|\bar{p}\| \leqslant \zeta_b \sqrt{V_2(0)} + \|\bar{p}_d\| \tag{45}$$

for some positive constant $\zeta_b \in \mathbb{R}$, where $\bar{p}_d(t) \in \mathcal{D}$ based on Theorem 1. To ensure that $\bar{p}(t) \in \mathcal{D}$, the image space needs to be sized to account for the effects of $\zeta_b V_2(0)$. Based on (38), $V_2(0)$ can be made arbitrarily small by increasing k_2 and initializing $\bar{p}_d(0)$ close or equal to $\bar{p}(0)$.

5 Conclusions

A path planner is developed based on an image-space NF that ensures the desired image trajectory converges to the goal position while also ensuring the desired image features remain in a visibility set under certain technical restrictions. An adaptive, homography-based visual servo tracking controller is then developed to navigate the camera-in-hand pose along the desired trajectory despite the lack of depth information from a monocular camera system. The path planner and the tracking controller are analyzed through a Lyapunov-based analysis. Simulation results are provided to illustrate the performance of the proposed approach.

References and Appendices are available in the following technical report

J. Chen, D. M. Dawson, W. E. Dixon, and V. Chitrakaran, "Navigation Function Based Visual Servo Control," Clemson University CRB Technical Report, CU/CRB/9/9/04/#1, http://www.ces.clemson.edu/ece/crb/publictn/tr.htm, Sept, 2004.