Pergamon

Mathl. Comput. Modelling Vol. 24, No. 5/6, pp. 79–102, 1996
Copyright©1996 Elsevier Science Ltd
Printed in Great Britain. All rights reserved
0895-7177/96 $15.00 + 0.00

PII: S0895-7177(96)00116-1

Camera Modelling for Visual Servo Control Applications

B. Bishop
Coordinated Science Lab, University of Illinois, Urbana, IL 61801, U.S.A.

S. Hutchinson
Elect. and Comp. Eng., University of Illinois, Urbana, IL 61801, U.S.A.

M. Spong
Coordinated Science Lab, University of Illinois, Urbana, IL 61801, U.S.A.

Abstract—When designing a visual servo system, it is important to have a complete and accurate model of the imaging process. Unmodelled imaging dynamics may play an important role in the stability and performance of such systems. In this paper, we present a detailed camera model which can be used in the design and analysis of visual servo systems.

Using the free-standing acrobot as a testbed, we analyze the effects of unmodelled imaging dynamics on visual servo control systems. We show that certain camera parameters strongly influence the performance of this system, and that accurate modeling is necessary for proper selection of imaging hardware.

1. INTRODUCTION

To perform effectively in real-world settings, robots must be able to plan and execute tasks in the presence of uncertainty. Typical sources of uncertainty in a robotic work cell include limited sensing accuracy, errors in dynamic models, and discrepancies between geometric object models and physical objects (including the parts to be manipulated and the robot itself). Because of this, the application of robotic technology to manufacturing problems has typically been restricted to situations in which uncertainty can be tightly controlled (for example, by using specialized fixturing devices).

One approach to dealing with the inherent uncertainty in a robot work cell is the use of physical compliance-based control schemes. For example, when a robotic manipulator is equipped with force sensing, it is possible to use hybrid position/force control to execute compliant motions [1,2]. The appealing characteristic of such motions is that they are constrained along one or more dimensions by the task geometry, thereby reducing the required precision in position information. The primary limitation of force feedback is that it can only be used to constrain motion along directions that are normal to workpiece surfaces at contact points. Pure position control must be used to control motions in directions that are tangent to the workpiece surface. Therefore, if the exact manipulator and goal positions are not known, the position-based control may fail to achieve the goal and hybrid position/force control is not sufficient.

One way to cope with the limitations of hybrid force/position control is to add vision sensing to the control servo loop. If the geometry of the imaging process is known, then the task geometry can be used to constrain the motions in the plane tangent to the workpiece surface using visual

This research was partially supported by the National Science Foundation under Grants IRI-9216428 and MSS-9212376, and by the Energy Power Research Institute (EPRI) under contract RP 8030-14.

Typeset by AMS-TeX

servo control. In recent years, the integration of computer vision with robot motion control has steadily progressed, from early look-and-move systems in which vision was used to recognize and locate an object prior to its manipulation [3,4], to current systems in which visual feedback is incorporated directly into the control loop [5–9]. This recent ability has made sensor-based robotics useful for a number of tasks where sensorless manipulation had previously failed, for example in welding [10–12].

A common characteristic of the visual servo control schemes reported to date is that the organization is hierarchical. The vision system is used in an outer "command loop," which generates reference inputs to an inner "robot control loop" (e.g., [5–9, 13–17]). This arrangement, which we shall refer to as a *dual-loop* visual servo controller, is illustrated in Figure 1. In dual-loop controllers, the vision loop typically runs at a frequency much lower than that of the robot controller. This difference in sampling rates is typically due to limitations of the vision system, which include limits on the sampling time for vision hardware and the computing time required by various vision algorithms.

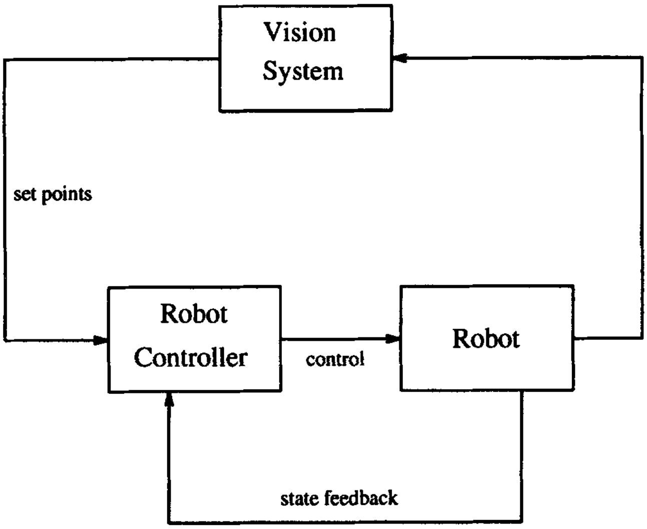

Figure 1. Dual-loop visual servo.

By keeping the vision sensing outside the servo level control loop, these hierarchical control schemes possess certain inherent robustness properties. The internal robot control loop is designed to guarantee that the system is stable, and relies on the vision system only for trajectories and set-points to track. Such robustness may be achieved at the expense of performance, in that the dynamics of the environment, including the vision system, are outside the feedback loop, and hence, cannot be compensated for. An alternative controller architecture is illustrated in Figure 2, which we shall refer to as a *direct* visual servo system.

In a direct visual servo system the vision system directly feeds back state information (instead of reference inputs) to the robot controller. Such an architecture may be advantageous in a variety of settings, and may in fact possess better overall qualities than the dual-loop systems described above for certain applications [6]. For example, in a *task space feedback linearization scheme* [18,19], one can define an output error as the difference between the position of the object to be tracked (environment) and the robot end-effector. This error signal can be used directly in the output feedback control scheme. In this case, the only means of sensing the object may be through the vision system.

In order to design robust direct visual servo control systems, it is important to fully understand the interactions between the image formation process and state estimation, where the state can be a combination of standard robotic state data and information about the environment. To this

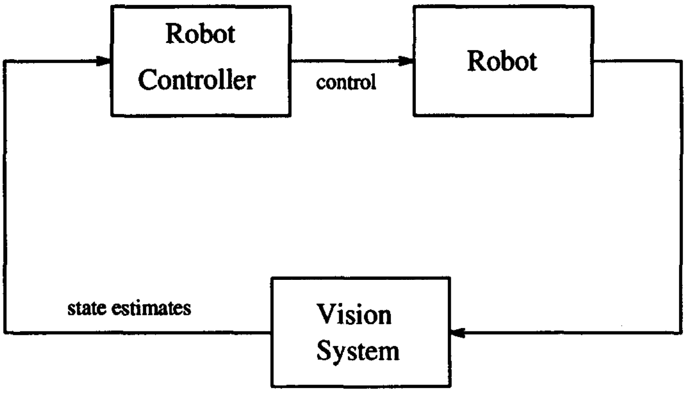

Figure 2. Direct visual servo.

end, we present a detailed model of the image formation process, including a set of parameters that characterize the departure of the system from the ideal (which in the computer vision literature is typically considered to be an ideal pinhole lens system). To date very little effort has been directed at the analysis of the relationship between distortions in the imaging process and the performance of visual servo systems [20], a fact that could be attributable to the nature of the dual-loop approaches currently in favor, wherein errors produced by imaging distortions can easily be overshadowed by the more prevalent effects of the set-point generation approach and the associated delays. Although our work is done specifically in the context of direct visual servo systems, the results may also benefit designers of dual-loop visual servo systems.

The remainder of the paper is organized as follows. In Section 2, we discuss the nature of the image formation process, including possible distortions and aberrations for a specific class of visual sensors, namely, the Charge Coupled Device (CCD) camera. We discuss a complete camera model and parameterize a set of disturbances that can affect visual data extracted from such a camera. In Section 3, we present a brief description of a robotic system, the free-standing acrobot, and a visual state estimation algorithm. In Section 4, we present a simulation study of direct visual servo of the free-standing acrobot under a variety of conditions. Discussion of the results are found in Section 5, and the conclusions are contained in Section 6.

2. THE IMAGING PROCESS

In this section, we present a detailed model of the imaging process for standard CCD cameras. We begin with a discussion of geometric issues, including projective geometry and lens distortions. Following this, we discuss photometric issues such as defocus, blur, etc. Finally, we present a brief discussion of the temporal aspects of the imaging process. The models that we present here will be used extensively in the analysis and empirical investigations of Section 4.

2.1. Geometric Aspects of Image Formation

In most existing visual servo systems, it is assumed that the imaging geometry can be adequately modelled as an ideal *pinhole lens*, as shown in Figure 3. With this model, a scene point whose coordinates in the camera frame are (x_c, y_c, z_c) projects onto the image plane as follows

$$u_i = f\left(\frac{x_c}{z_c}\right), \tag{1}$$

$$v_i = f\left(\frac{y_c}{z_c}\right), \tag{2}$$

where f is the focal length of the camera lens, and the subscript i indicates the ideal image coordinates, with no distortion. For even a simple lens imaging system, this model is fairly naive.

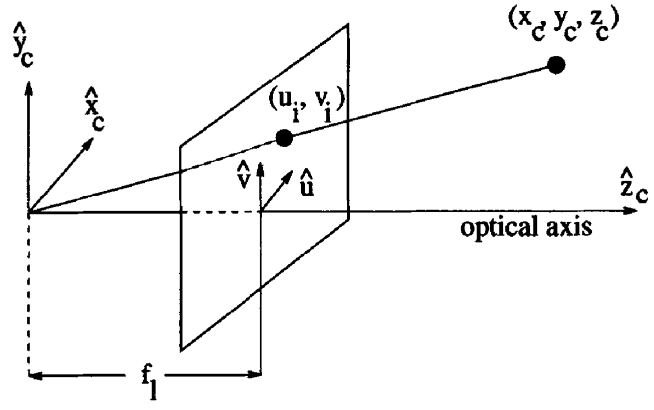

Figure 3. Pinhole lens with associated coordinate systems.

In a real imaging system, lens distortions or aberrations can affect the geometric correspondence between points in the camera's field of view and points in the image. Several camera calibration schemes seek to identify the set of lens distortions for off-the-shelf cameras [21–24].

We have chosen the work of [22] as the basis of our model, due to its completeness and variety of distortion parameters. The three types of distortion that are documented there are *radial distortion*, *decentering*, and *thin prism distortion*. We now give a brief overview of each of these distortions and discuss the physical characteristics that generate them in a real camera system. Systematic derivations of the resulting distortion equations can be found in [22] and its associated references.

Radial distortions are perhaps the most common and familiar to the image processing and optics communities. Radial distortions cause a displacement of image points toward (pincushion distortions) or away from (barrel distortions) the optical axis, which intersects the center of the image plane (see Figure 3). Radial distortion arises from flaws in the lens construction, typically in the curvature of one or more of the lenses in the system. While it cannot be corrected without the addition of complicated lens systems or replacement of the camera, it does not change over time. Once isolated, radial distortion can be considered a known parameter of the imaging system. The result of radial distortion is a displacement of $(\delta_{ur}, \delta_{vr})$ from the ideal image coordinates, given by

$$\delta_{ur} = k_1 u \left(u^2 + v^2\right) + O\left[(u,v)^5\right], \tag{3}$$
$$\delta_{vr} = k_1 v \left(u^2 + v^2\right) + O\left[(u,v)^5\right], \tag{4}$$

where k_1 is positive for barrel and negative for pincushion. We use the notation $O[(u,v)^n]$ to denote an n^{th} order term in the image coordinates (u, v).

Figure 4 shows a normalized vector field generated by barrel distortion, with a positive value for k_1. The small dots are the heads of the vectors, which represent the normalized distortion vector $[\delta_{ur}, \delta_{vr}]$ of the image plane point from which the vector originates. Negative values of k_1 produce a similar vector field that is aligned at all points toward the optical axis.

Decentering distortion occurs when the elements in a lens system are not aligned properly, so that the optical axes of the lenses may differ slightly. This type of distortion can easily appear after a camera system has been moved or disassembled and reassembled, even if it was not present earlier. The effect of decentering distortion, parameterized by (p_1, p_2), is a displacement of $(\delta_{ud}, \delta_{vd})$ from the ideal image coordinates, given by

$$\delta_{ud} = p_1 \left(3u^2 + v^2\right) + 2p_2 uv + O\left[(u,v)^4\right], \tag{5}$$
$$\delta_{vd} = 2p_1 uv + p_2 \left(u^2 + 3v^2\right) + O\left[(u,v)^4\right]. \tag{6}$$

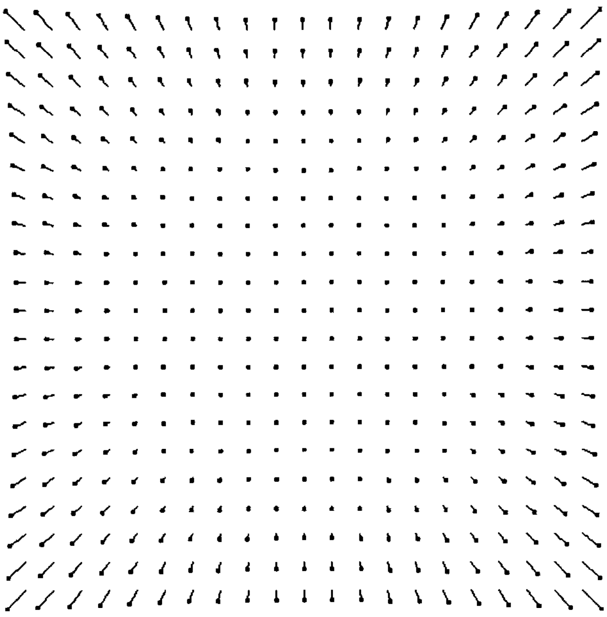

Figure 4. Vector filed distortions from a positive k_1.

Figures 5 and 6 show the vector fields of the distortion generated by positive values of p_1 and p_2. Negative values of p_1 would simply reflect Figure 5 about the vertical axis through the center point. Negative values for p_2 similarly reflect Figure 6 about the horizontal axis through the center point.

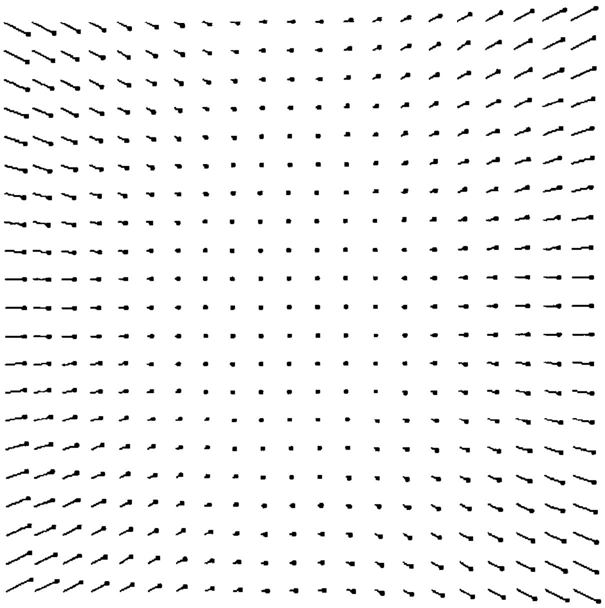

Figure 5. Vector field of distortions from a positive p_1.

Thin prism distortion results from improper lens design or from errors in the construction of the lens system. Typical examples include variations in the radius of curvature over a single side of a lens or a slight tilt of a lens in the array. This type of distortion can be modelled by adding a parameterized thin prism to the system. The resulting image plane displacement is given by

$$\delta_{up} = s_1 \left(u^2 + v^2\right) + O\left[(u,v)^4\right], \tag{7}$$

$$\delta_{vp} = s_2 \left(u^2 + v^2\right) + O\left[(u,v)^4\right], \tag{8}$$

where s_1 and s_2 are the parameters of the thin prism. Figures 7 and 8 show the distortion fields

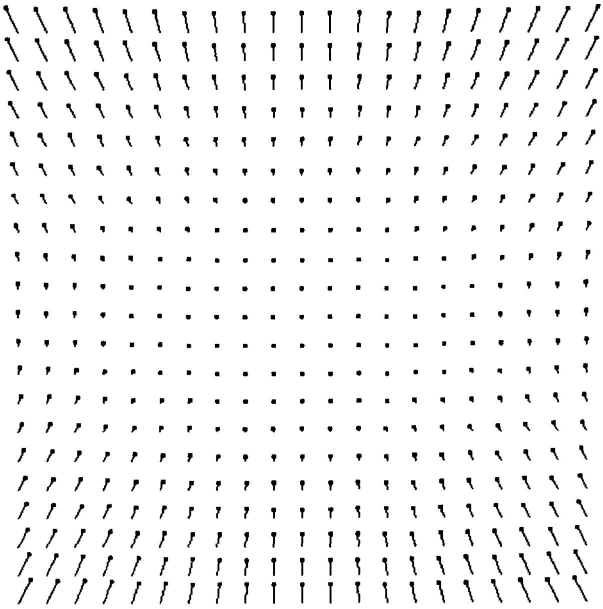

Figure 6. Vector field of distortions from a positive p_2.

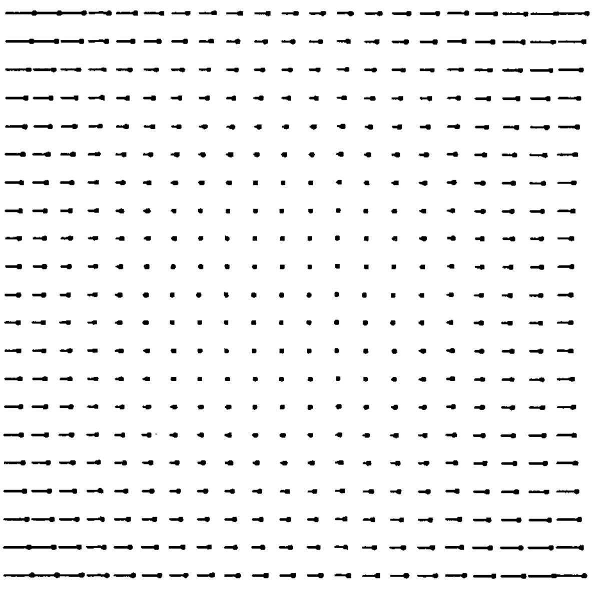

Figure 7. Vector field of distortions from a positive s_1.

for positive values of s_1 and s_2, respectively. Negative values of these parameters simply change the sign of the vectors, as can be seen from examining (7) and (8).

Combining the effects of these three types of distortion gives the following set of equations for total distortion of ideal image points (u_i, v_i):

$$u = u_i + \delta_{u_i r} + \delta_{u_i d} + \delta_{u_i p}, \tag{9}$$

$$v = v_i + \delta_{v_i r} + \delta_{v_i d} + \delta_{v_i p}. \tag{10}$$

We simplify our model by ignoring terms in the distortions that are above order three. In fact, it is known that an exact representation of these disturbances requires an infinite number of terms [25]. We use only the low-order terms since empirical evidence has suggested that ignoring these high-order terms is appropriate [21,22]. Further, we define the area of interest in the image plane to be $-1 \leq u, v \leq 1$, thereby making these terms much less significant. While this

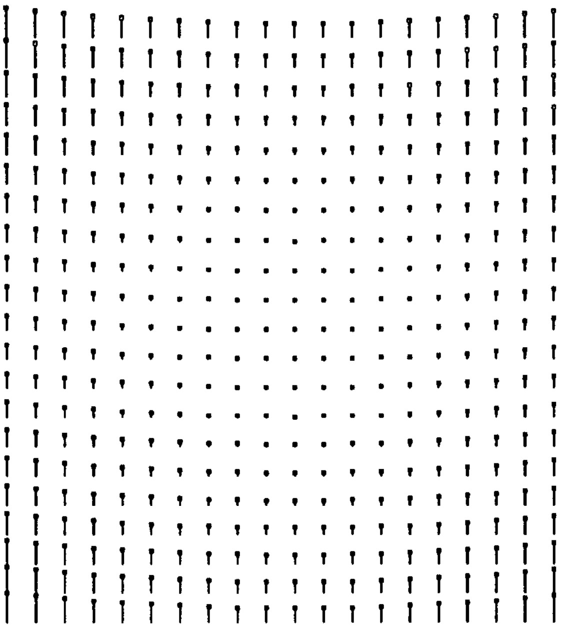

Figure 8. Vector field of distortions from a positive s_2.

restriction of interest relies on the image plane units chosen, all distortion parameters will be considered in $(\text{units})^{-1}$ or $(\text{units})^{-2}$, thereby inherently containing the scaling information.

2.2. The Photometric Aspects of Image Formation

To this point, we have been concerned with the geometry of the transfer of light from a scene to the image plane. We will now focus on the method by which this light is transformed into a discrete image in the camera. In doing so, we will describe a variety of parameters and disturbances that can affect the intensity levels of the pixels.

The image plane of a CCD camera is made up of an array of photosensors, such as photodiodes, coupled to charge storage devices, such as Metal-Oxide Semiconductor (MOS) capacitors. The output voltage level of a given photodiode element at any given time is proportional to the average intensity of incoming light over its entire sensing area, and tends to be a function of both the intensity of the light and its wavelength. A photosensor of any type has a characteristic *element sensitivity* function with respect to incoming light, and the nature of any image generated is based on this sensitivity function.

Camera sensitivity, as distinguished from element sensitivity, is the responsiveness of each sensing element to the light incident on the portion of the *lens* that would ideally focus onto that element. It is often the case that an imaging system shows excellent camera sensitivity near its optical axis, but that the lens system does not deliver proper amounts of incoming light near the edges of the CCD array, causing what should be identical intensities at the optical axis and the edge of the image to differ by several grey scale levels. Thus, camera sensitivity is a result of aberrations in the lens system, different from those geometric aberrations detailed in Section 2.1, and has no connection to the CCD array or its dynamics. Under this assumption, we model this disturbance as

$$I = I_0 \cos^{\beta}\left(\tan^{-1}\left(\frac{r}{f}\right)\right), \tag{11}$$

where I is the distorted image intensity, I_0 is the ideal, undistorted image intensity, $r = \sqrt{u^2 + v^2}$ image plane units, f is the camera focal length in image plane units, and β is a nonnegative integer.

If there is relative motion between the camera and the scene during image formation, the resulting image will be blurred. Under normal circumstances, the image blur is a simple function of the motion and the camera equations as defined above. In many cases, this blur will not present a problem, as the image plane motions of the features of interest may be very small over the sample interval. In addition, many CCD cameras can be electronically shuttered to provide an exposure time that is very small with respect to the frame rates, typically as short as one millisecond or less, reducing or eliminating blur. Specialized CCD cameras exist that have even lower exposure times, with frame rates as high as 10,000 frames per second, albeit with low resolution [26]. We shall define a parameter called the *exposure factor*, denoted E_t, which is the fraction of the sample period under which the shutter is open. By default we will assume this factor is 1.0. This parameter will play a significant role in the experimental dynamic analyses of Section 4.

Another problem that can arise in most types of imaging systems is defocus, which is often confused with motion blur. If a camera system is not in proper focus, the image will appear blurred, even with no motion in the scene. Defocus can be represented by a two-dimensional convolution of the image intensity profile with a radially symmetric function known as the *point spread function*. We will assume that defocus can be modelled by a sequence of N convolutions with a mask such as the one in Figure 9. The parameters f_0, f_1, f_2 and the number N are the parameters for the defocus.

f2	f1	f2
f1	f0	f1
f2	f1	f2

Figure 9. Mask for discrete convolution in defocus simulation.

Finally, as with any measured value in the real world, the intensity of an image pixel is subject to noise in the image formation process. We model image noise as an additive term with a zero mean Gaussian distribution and a standard deviation of σ grey levels.

2.3. Temporal Aspects of Image Formation

The standard video frame rate is thirty frames per second, but many CCD cameras employ an interlaced scan method by which a single $N_x/2 \times N_y/2$ frame is available at twice this rate. In addition, cameras designed for slow motion tend to have sample rates which greatly exceed the standard 30 Hz. We will denote the sample rate of the simulated camera by ω_{cam} (samples/sec).

The final parameter we consider depends not only on the camera system, but on the algorithms and hardware used to derive information from the images. There is a finite time interval T_{out} required by the camera to output the image data to external devices, which then require some interval T_{calc} to calculate or extract the desired data to be used by the control system. We will combine these two delays into a single parameter called T_{delay}. Delays of this sort are both intrinsic and extrinsic as they depend on the camera system as well as the type of calculations required by the visual servo system. Often, this type of delay is ignored in visual servo system design; this problem will be considered more completely in Section 4.

3. EXPERIMENTAL SETUP

To evaluate the influences of the various imaging system parameters, we have designed a direct visual servo controller for the free-standing acrobot. In this section, we will briefly describe the free-standing acrobot, its dynamics, and our controller.

3.1. The Free-Standing Acrobot

The free-standing acrobot, illustrated in Figure 10, is a two-link, planar robot with a single actuation between the links. The end of the first link, which is to be in contact with the ground plane, is rounded and has no moving parts. The robot is free-standing and can function on any smooth surface. In the literature, the free-standing acrobot is also referred to as the *rolling* acrobot, which can sometimes cause confusion. For the remainder of this paper, we will refer to the free-standing acrobot simply as the acrobot.

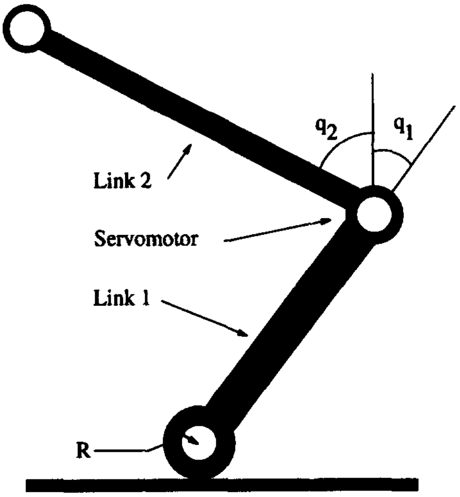

Figure 10. The free-standing acrobot.

The dynamic equations of the acrobot as described are

$$M(q)\ddot{q} + c(q,\dot{q}) = u, \tag{12}$$

where $q = [q_1, q_2]^{\mathrm{T}}$ and

$$M(q) = \begin{bmatrix} \theta_1 + \theta_2 \cos(q_1) & \theta_7 \cos(q_2) + \theta_8 \cos(q_1 - q_2) \\ \theta_7 \cos(q_2) + \theta_8 \cos(q_1 - q_2) & \theta_5 \end{bmatrix},$$

$$c(q,\dot{q}) = \begin{bmatrix} -\theta_9 \sin(q_1) \\ -\theta_{10} \sin(q_2) \end{bmatrix}, \quad \text{and} \quad u = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \tau.$$

The value τ is the input torque. The quantities $(\theta_1, \theta_2, \theta_5, \theta_7, \dots, \theta_{10})$ are the dynamic parameters of the system. We define m_1 and m_2 as the masses of links one and two, respectively, l_{c1} and l_{c2} as the distance from the axis to the center of mass of links one and two, respectively, l_1 as the length of link one, R as the radius of the "drum" on the end of link one, and J_1 and J_2 as the moments of inertia of links one and two, respectively. The dynamic parameters of the system are given in Table 1. The nonsequential numbering of the dynamic parameters allows correspondence between these dynamic equations and those of [27].

When the input torque between the two links is zero, the device has an unstable equilibrium point corresponding to the vertical position. For each value of input torque τ, the device has another unstable equilibrium point. The set of all such points defines an unstable equilibrium

Table 1. Dynamic parameter definitions for the acrobot.

Name	Definition	Type
θ_1	$m_1R^2 + m_2{l_1}^2 + m_2R^2 + J_1$	Inertial
θ_2	$2m_1Rl_{c1} + 2m_2Rl_1$	Inertial
θ_5	J_2	Inertial
θ_7	m_2Rl_{c2}	Inertial
θ_8	$m_2l_1l_{c2}$	Inertial
θ_9	$m_1gl_{c1} + m_2gl_1$	Gravity
θ_{10}	m_2gl_{c2}	Gravity

manifold in the robot's state space, shown in Figure 11 as a function of the state (q_1, q_2) given in radians. The equation for this manifold is

$$q_1 = -\arcsin\left(\frac{\theta_{10}}{\theta_9}\sin(q_2)\right). \tag{13}$$

The manifold shown is for $(\theta_9, \theta_{10}) = (4.779, 1.880)$, which will be used in the simulations that follow.

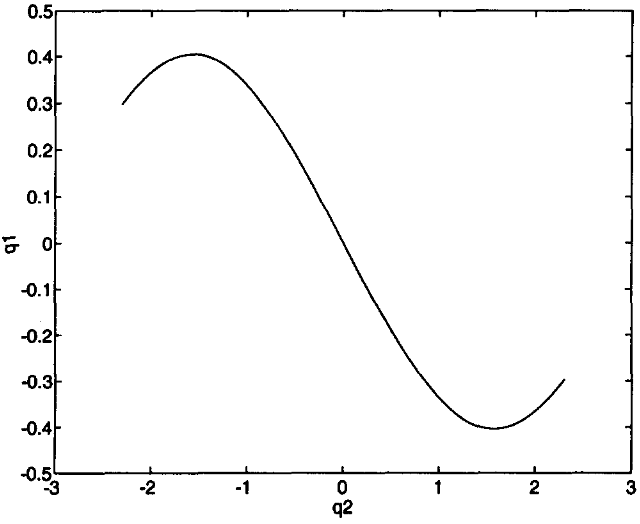

Figure 11. Equilibrium manifold for the acrobot.

The goal of a control system design is to stabilize the acrobot in a region of this manifold. Trajectories to be tracked or set-points to be followed would then be generated as a function of the position on the manifold. For purposes of this work, let the manifold be parameterized by the variable z, with positive values corresponding to positive q_2 values. A value of $z = 0.5$ corresponds to a distance along the manifold from $q_2 = 0$ of about 75% of the total length of that arm of the manifold. Note that in Figure 11, the manifold only extends to approximately $q_2 = 2.35$ radians, which is the point at which the links are assumed to contact one another. The origin of the z-parameterization of the manifold lies in the original controller design of [27]. A quantitative examination of the control algorithm is beyond the scope of this work. In order to shed some light on the analyses that follow, we will offer a brief, qualitative discussion of the controller. Details can be found in [27].

A standard approach in nonlinear control is that of *feedback linearization*. Under certain conditions on the dynamics of a given system, a nonlinear state transformation and state feedback can be found such that, when applied to the system, its dynamics become linear. Making a nonlinear system behave linearly is desirable due to the broad range of well-known linear control algorithms available.

It can be shown that the nonlinear dynamics of the acrobot cannot be transformed into linear system dynamics through the use of state feedback. Instead, we use a technique known as *pseudolinearization* [28]. Through the use of a coordinate transformation and nonlinear feedback control law, a system can be forced to have a Jacobian linearization along some equilibrium manifold that is independent of the location along that manifold. The exact form of the control can be modified to exhibit specific behaviors that are desired, such as system tracking along the equilibrium manifold.

This approach, unlike feedback linearization, can be successfully applied to the acrobot. The control law is complex, and involves extensive use of B-splines for numerical calculation. The important characteristics for purposes of this work are that the final control algorithm is designed to force the acrobot to track positions on its unstable equilibrium manifold, and that the system is only shown to be globally uniformly ultimately bounded. This indicates that we can only prove that the system will asymptotically converge to a neighborhood of the desired equilibrium point, but may be at any point inside that neighborhood at any time after its convergence. Further, while the theoretical work is global, the physical system has additional constraints, in that it is possible for the links to collide. Thus, for our purposes, the control result holds only in a local region of attraction of the equilibrium manifold.

The controller that we have implemented requires full state feedback, i.e., $(q_1, q_2, \dot{q}_1, \dot{q}_2)$. However, in our implementation, $\dot{q}_1$ and $\dot{q}_2$ are calculated from successive observations of (q_1, q_2) using first order difference approximations. Therefore, in the context of state estimation below, we will refer to the couple (q_1, q_2) as the "state" of the acrobot.

Because the first link of the acrobot is unpinned and has no moving parts at its contact point with the ground plane, measurement of the state of the robot via traditional means (e.g., using encoders) is not possible. At present, there is no functioning control scheme for state estimation of this system that does not involve hardware modifications to the acrobot itself. The existing scheme of [27] relies on an assumption that the acrobot has only rolling contact with the ground plane.

The pseudolinearizing control algorithm of [27] can be combined with visual state estimation to comprise a direct visual servo control system. Because the control is being attempted along an unstable manifold, and the controller has only been proven globally uniformly ultimately bounded, errors in the state estimates are of paramount concern. Therefore, this system provides a suitable test bed for direct visual servo control.

3.2. State Estimation

The role of the vision system in our direct visual servo approach is to determine the state of the robot, (q_1, q_2), as shown in Figure 10. In order to ease the task of state estimation, we have made minor cosmetic modifications to the robot. In particular, states are estimated by determining the centroids of images of circles that are inscribed on the robot's links. Such techniques have been used in many recent visual servo systems, e.g., [7,14]. Many other feature tracking algorithms have been proposed in the literature, e.g., [9,29,30].

Our system is configured so that the robot lies in a plane perpendicular to the optical axis of the (fixed) camera. Feature circles are placed at the joint and at the tips of the first and final links. Each camera image is thresholded at a user-defined level, generating a binary image that ideally consists only of images of the inscribed circles. Centroids can easily be calculated from such binary images.

Assuming knowledge of the correspondence between image circles and the actual inscribed circles on the robot, calculation of the state is a simple task. In order to determine this correspondence, it is assumed that the sample rate of the vision system is small enough that the maximum displacement of each circle between images is less than one-half the smallest separation between any two circles. Use of the robot's Jacobian allows the designer to be aware of the largest possible Cartesian space velocity of each inscribed circle based on the possible generated torque τ, and to modify positioning or make other arrangements if correspondence might become a problem in certain configurations of the inscribed circles. Assuming an initial knowledge of the correspondence, we can determine which circles in subsequent images identify with circles in the original image and thereby with the inscribed circles, by straightforward feature tracking.

The problem of noise necessitates the introduction of two new parameters for the sensing system. N_{min} and N_{max} are the smallest and largest allowable areas, in pixels, of image regions that will be identified as the images of the feature circles. This visual algorithm can easily be implemented in standard video hardware (e.g., the Datacube MaxVideo system).

The robot states (q_1, q_2) are calculated from each image frame and passed, along with the calculated derivatives ($\dot{q}_1$, $\dot{q}_2$), to the controller of [27]. The camera is assumed to be a perfect pinhole lens. Any lens distortions or other imaging aberrations are then unmodelled imaging dynamics. In Section 4, we consider the errors in the state estimates generated by the unmodelled image dynamics and their effects on control system performance.

4. SIMULATIONS AND ANALYSES

In this section, we will consider the effects of the various imaging dynamics and distortions discussed in Section 2 on both the state estimation routine as described in Section 3.2, and on the stability of the entire direct visual servo system. Details of the simulation can be found in [27,31].

For the following simulations, we consider the field of view of the fixed camera to be a square of 20 × 20 units in the world space. The acrobot stands on the (horizontal) ground plane in a plane perpendicular to the optical axis of the camera. When both links are vertical ($q_1 = q_2 = 0$) and no distortions are present, the acrobot covers approximately 70% of the total height of the image plane. Initial placement of the acrobot is such that in its vertical position it lies along the camera's vertical (v) axis. Since the acrobot moves across the ground plane as q_1 changes, an initial configuration with a nonzero q_1 will have a horizontal displacement of the contact point from $u = 0$.

The dynamic parameters of the simulated acrobot are given in Table 2. The lengths of the links are 6.19 units for the first link and 6.9 units for the second link, with a base radius of 1.0 unit. The inscribed circles each have a radius of 0.2 unit.

Table 2. Dynamic parameters for acrobot simulations.

Name	Value
θ_1	0.20386 kg·m^2
θ_2	0.05867 kg·m^2
θ_5	0.11 kg·m^2
θ_7	0.01437 kg·m^2
θ_8	0.0989 kg·m^2
θ_9	4.779 Nt
θ_{10}	1.880 Nt

The simulated camera implements the model derived in Section 2.2. The output is assumed to be a 512 × 512 pixel matrix quantized to 256 grey levels. Unless otherwise stated, the exposure factor E_t was assumed to be small, on the order of 0.01, and ω_{cam} was 30 Hz. It was determined that the inscribed circles with radius 0.2 unit in the world coordinate frame would cover approximately 75 pixels. There was no provision made for specularity, shadowing or reflection, nor for nonhomogeneity of lighting sources.

4.1. State Estimation

In this section, we will consider the effects of various disturbances on estimation of the state of the acrobot. We will consider all of the disturbances detailed in Section 2.2 except for delay, which is unimportant for purposes of state estimation. For the following simulations, we will consider the acrobot as it moves along its unstable manifold. In all of the simulations, the acrobot was initially placed with $(q_1, q_2) = (0, 0)$, which corresponds to the unactuated equilibrium point. The acrobot was then positioned at a variety of points along its unstable manifold and the estimation error was measured over the entire manifold. All plots are given as error versus value of q_2, which intuitively parameterizes the unstable manifold. We have chosen q_2 instead of q_1, since the manifold is parameterized by a function $q_1 = f(q_2)$, where f^{-1} does not exist. All values are given in radians and estimation errors are of the form $q_{\text{actual}} - q_{\text{estimated}}$.

Using a discrete system such as a CCD camera to estimate a continuous value, such as the state of the acrobot, causes quantization error both in the intensity value and in the image plane coordinates of the centroids of the inscribed circles. Figure 12 shows the corresponding state estimation error for the case of no distortions.

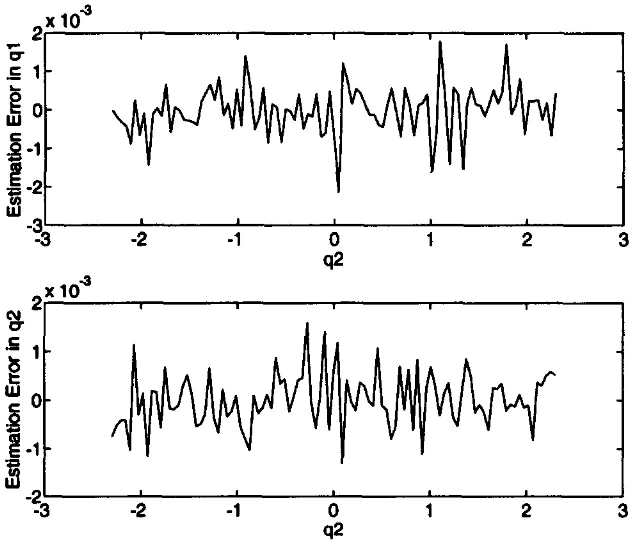

Figure 12. Error in visual state estimation with no distortions.

4.1.1 Blur, focus, camera sensitivity, and noise

The state estimates calculated when blurring was present (i.e., when the acrobot was in motion and E_t was near one) tended to give the value of the states at the midpoint of the exposure interval unless the angular velocities of the links were too high, in which case the system lost identification as the feature blurred out to a grey scale level below the threshold. We have experimentally determined that 0.15 unit per second is the velocity along the unstable equilibrium manifold at which the system will lose identification, assuming E_t has a value of one, so that the exposure interval is equal to the sample period, 1/30 second.

Experimentation has demonstrated that the amount of defocus characteristic of typical systems is negligible for our purposes. Further, it is noted that poor camera sensitivity (i.e., high values of β) effectively reduces the usable region of the image plane for thresholding-based techniques. As the features move away from the optical axis, the pixels in the feature image drop in intensity. At some point, they cross the threshold value and are mapped to zero. We will assume that the features of interest lie in the region of the image plane in which their image pixel values remain above the threshold. Under this assumption, camera sensitivity has little effect on the state estimates. Simulations and experiments have shown that Gaussian noise, even with high variance, has little effect on the estimation error of the sensing strategy we have chosen. Details are given in [31].

4.1.2. Lens distortions

We present, in this section, an analysis of the effects of the five lens distortion parameters (k_1, p_1, p_2, s_1, s_2). Figures 13–17 show the errors in the estimates of (q_1, q_2) along the unstable equilibrium manifold for each of the lens distortion parameters at a value of 0.1, with all other distortions equal to zero (including zero blur). In all of the figures, the estimation error in q_1 is denoted by a solid line and estimation error in q_2 by a dashed line.

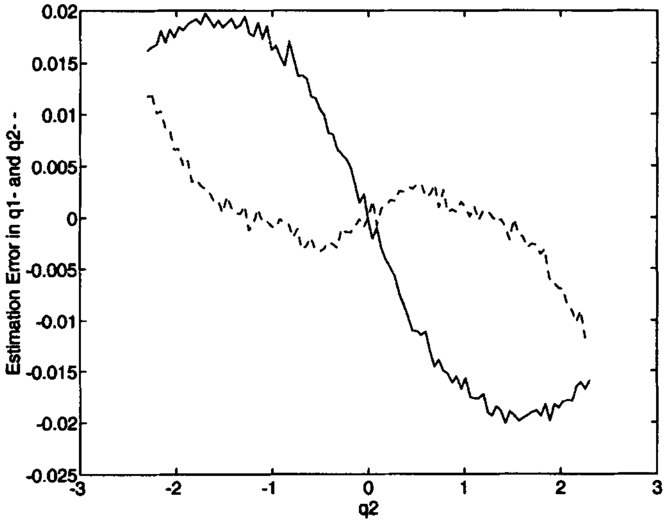

Figure 13. State estimation error (q_1 solid, q_2 dashed) under radial distortion (k_1 = 0.1).

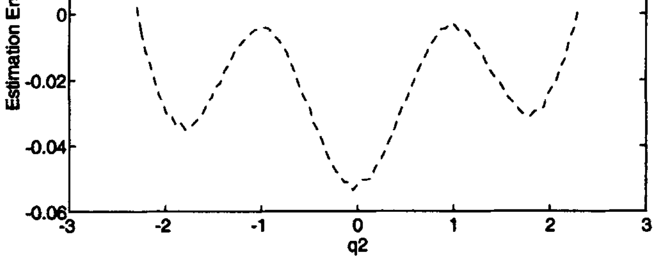

Figure 14. State estimation error (q_1 solid, q_2 dashed) under decentering distortion (p_1 = 0.1).

Changing the sign of s_1 or p_1 leaves the shape and magnitude of the corresponding error profile unchanged, but reflects it about the $q_2 = 0$ axis. Changing the sign of k_1, s_2, and p_2 results in almost unchanged error curve shapes as well, except that the sign of the errors are reversed. Like p_1 and s_1, a sign reversal on k_1 leaves the magnitude of the estimation error almost unchanged. Negative values of p_2 and s_2, however, result in estimation errors for q_1 that are smaller in absolute value than those corresponding to equivalent positive parameter values. The estimation error for q_2 in these cases increases over a portion of the unstable manifold near $z = 0$, but decreases near the extremes of the manifold.

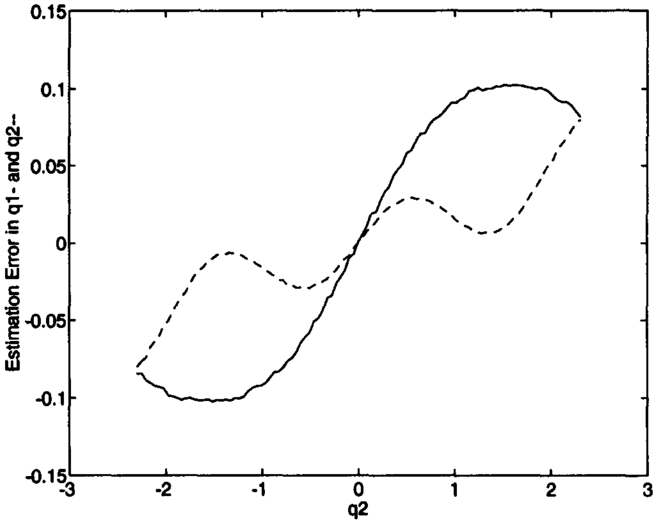

Figure 15. State estimation error (q_1 solid, q_2 dashed) under decentering distortion (p_2 =

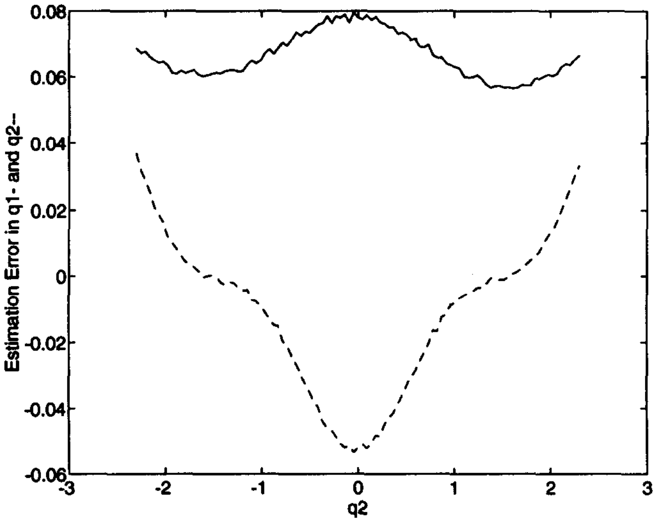

Figure 16. State estimation error (q_1 solid, q_2 dashed) under thin prism distortion ($s_1 = 0.1$).

The error in the state estimates has a characteristic shape for each type of distortion. Varying the magnitudes of any of the lens distortion parameters causes a proportional change in the estimation errors due to that distortion, but no change in the characteristic shape. This is to be expected, since (3)–(7) are linear in the distortion parameters.

We have determined estimates for the approximate proportionality constants relating the absolute value of each parameter to the maximum state estimation error over the unstable equilibrium manifold. For each distortion parameter, there are corresponding K_{q_1} and K_{q_2} values relating the parameter's value to the maximum absolute value of the estimation error for q_1 and q_2, respectively. These values are given in Table 3.

We note that, for the simulations of $p_2 = 0.1$ and $s_2 = 0.1$, certain parameters used in our feature segmentation algorithm have to be tuned, specifically $N_{\min}$ and $N_{\max}$. The need for this reevaluation of parameters could be seen in the amount of overall distortion. In a real system, the necessity for such a tuning of the parameters would be detected during setup. In fact, the distortion in these cases was so strong that it is unlikely the designer would ever have considered the simulated camera as a viable sensor.

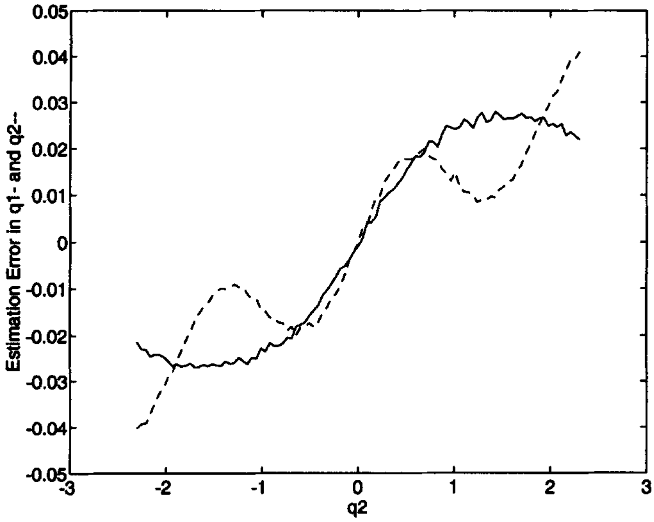

Figure 17. State estimation error (q_1 solid, q_2 dashed) under thin prism distortion ($s_2 = 0.1$).

Table 3. Proportionality constants for estimation error.

Parameter	K_{q_1}	K_{q_2}
k_1	0.2	0.1
p_1	0.8	0.5
p_2	1.0	0.8
s_1	0.8	0.5
s_2	0.3	0.4

It is almost always the case that, with image plane units of cm, $|p_1|, |p_2|, |s_1|, |s_2| \ll 0.1$. However, even with the distortions on the order of 0.1, the state estimates were consistently within 0.1 radian of the correct value. For values of these distortions that are not readily observable by the human eye in images generated by the camera, the errors in the state estimates will be much smaller.

4.1.3. Composite effects

In this section, we consider state estimation when a variety of aberrations and disturbances are simultaneously present in the lens system. The following simulation uses values that could easily be encountered in a real camera system.

We allow noise with $\sigma = 15$ grey levels and a defocus of $(f_0, f_1, f_2) = (5/9, 2/27, 1/27)$ with $N = 2$ (which is rather excessively defocused). We do not consider blur. The lens distortion parameters used are $(k_1, p_1, p_2, s_1, s_2) = (0.2, 0.002, 0.002, 0.02, 0.01)$, which are on the order of those experimentally determined in [22]. The resulting estimation error over the unstable equilibrium manifold is given in Figure 18. Estimation errors are on the order of 0.05 radian.

4.2. Control Simulations

In Section 4.1, we considered the effects of imaging distortions on the state estimates. We now consider the effects of these state estimation errors on the stability and performance of the entire direct visual feedback system, thus relating imaging distortion to control system performance.

In Sections 4.2.1 and 4.2.2, we will consider the effects of only those distortions that were seen to have a strong impact on the state estimates. All simulations were begun with an initial total state of

$$[q_1, q_2, \dot{q}_1, \dot{q}_2]^{\mathrm{T}} = [-0.05, 0.1, 0, 0]^{\mathrm{T}}. \tag{14}$$

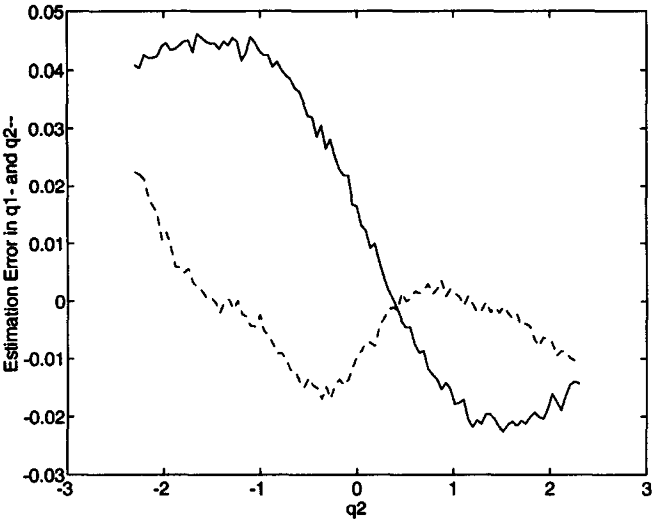

Figure 18. State estimation error (q_1 solid, q_2 dashed) under all aberrations.

Figure 19 shows the performance of the acrobot under direct visual servo control as it is given a z_{desired} function to track on its unstable equilibrium manifold. No distortions were present, and the exposure was assumed to be instantaneous at the end of each sample interval. No delay from any calculation or image processing was present. The four graphs in Figure 19 represent the important state and control information over the simulation interval, and provide a point of comparison for the following simulations. Simulations of the acrobot under full state feedback are shown in Figure 20. In all of the control simulations, the system's tracking behavior is displayed in the upper left subgraph. The signal to be tracked, z_{desired}, is the solid line, while the system's actual position on the unstable equilibrium manifold is given by the dashed line.

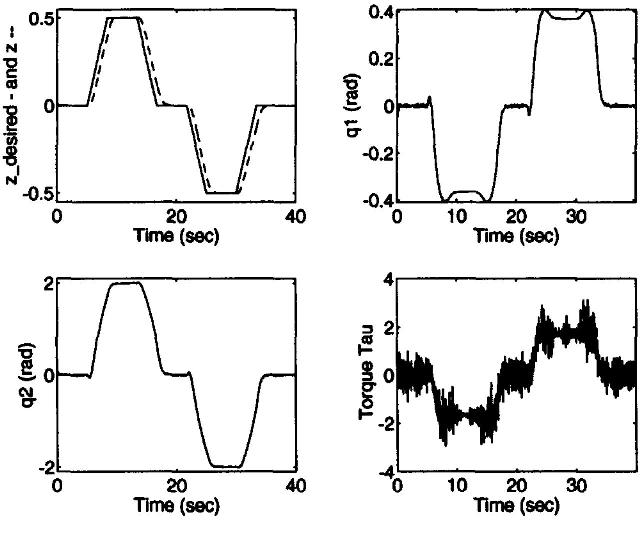

Figure 19. Visual servo of the acrobot with no distortions (see text).

We see that neither quantization errors (and their effects on state estimates) or the first order approximations for $\dot{q}_1$ and $\dot{q}_2$ have noticeable effect on system stability. We note that the results of the full state feedback simulation are almost identical to those of the visual servo case, except for the lack of significant chattering in the control signal. The chattering in the visual servo case can be accounted for by the discrete nature of the state estimates in combination with the first order velocity approximations.

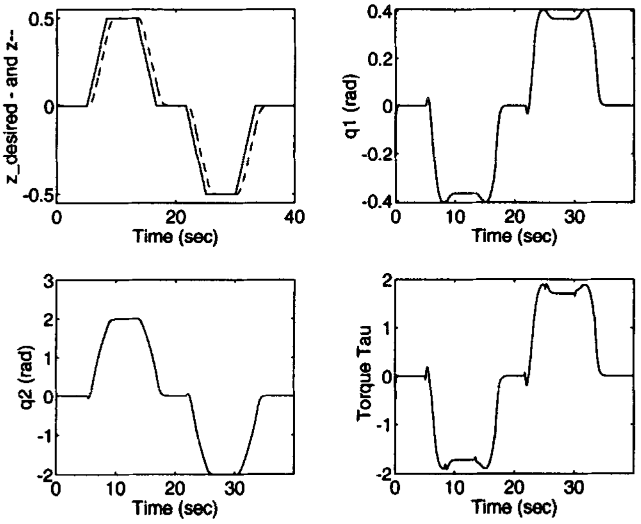

Figure 20. Rolling acrobot under full state feedback control (see text).

4.2.1. Control of acrobot under system delay and blur

There are several sources of delay in the feedback loop of this direct visual servo system. There is an inherent delay of approximately one-half of the period during which the shutter is open (or $0.5E_t/\omega$ seconds). This is a result of the integrating nature of CCD camera elements. Processing takes additional time that may or may not be significant.

To determine if this delay was significant to our control system, a simulation was performed under a delay of one full period, or 1/30 of a second, together with the delay inherent when E_t has a value of 1.00. Thus $T_{\text{delay}} = 1.5T$ where T is the sample period. Further, noise was present with $\sigma = 15$ grey levels in addition to a defocus of $(f_0, f_1, f_2) = (5/9, 2/27, 1/27)$ with $N = 1$. No other lens distortions were present. At 30 Hz, this system became unstable, even for very small $\dot{z}_{\text{desired}}$. At 60 Hz, the system was stable. This indicates that our computations should not require a full sample period at 30 Hz—the delay is too great. The inherent delay of one-half a sample period (due to blur) at 30 Hz did not cause instability, nor did an additional delay of one-half the sample period. This indicates that our vision and control calculations should be able to be carried out in 1/60 of a second. For the algorithm described in Section 3.2, this is an acceptable restriction.

4.2.2. Control of acrobot under lens distortions

In the following simulations, the values of the distortion parameters were chosen so that the maximum state estimation error over the entire unstable manifold would be approximately 0.01 radian. No defocus, noise, or blur were allowed, so that the effects of the individual distortions could be more clearly evaluated.

Unlike the case of state estimation, we are concerned here with distortions of both signs for all cases except p_1 and s_1, which simply reverse their effects with respect to $z = 0$ on the unstable equilibrium manifold (see Figures 5 and 7). We are concerned with the effect of the errors in state estimates on the performance and stability of the system. Note that, if the device reaches $q_2 = 2.35$ radians, the two links have contacted one another and the system is assumed to have become unstable.

The following simulations show the tracking error in z, the values of q_1 and q_2 with respect to time, and the input torque for various values of the lens distortion parameters. When we consider the state estimation error graphs given in Section 4.1, Figures 13–18, we can see that the error graphs for p_1 and s_1 are symmetric with respect to $q_2 = 0$, while those of the remaining distortions are antisymmetric.

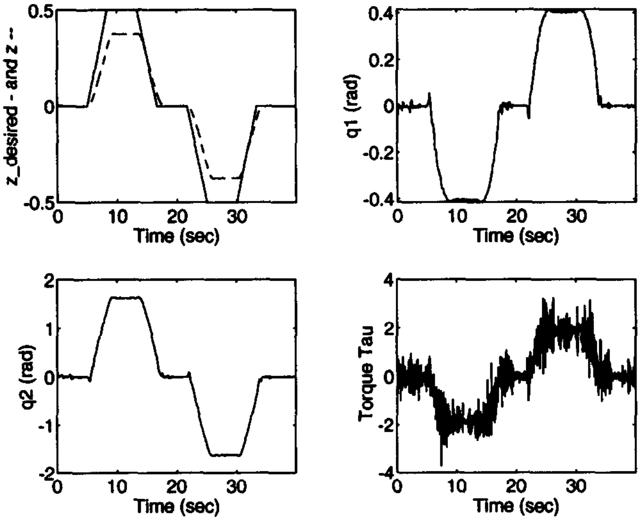

Figure 21. Visual servo with $k_1 = 0.1$ (see text).

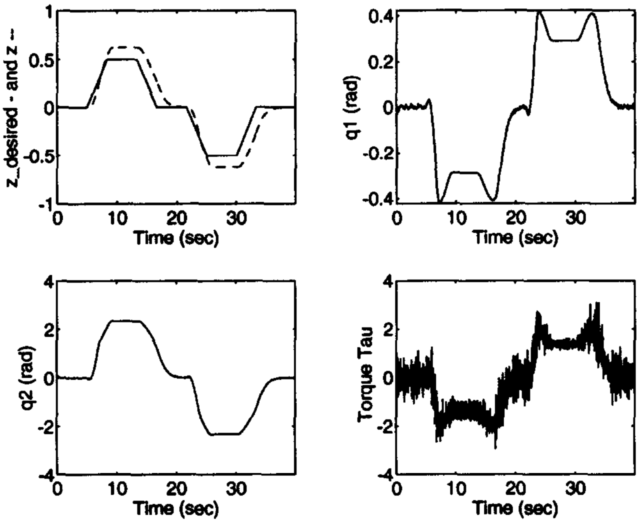

Figure 22. Visual servo with $k_1 = -0.1$ (see text).

In Figures 21 and 22, we see the simulations for $k_1 = \pm 0.1$. We note that the very small errors in the state which are produced by the vision system, together with the erroneous values of velocities, cause a significant steady-state error in z. The magnitude of this error is dependent on the portion of the manifold that the system is to track, since the state estimation error varies greatly over the workspace when lens distortions are present. The steady-state error for $z_{\text{desired}} = 0.5$ is approximately 0.1 in absolute value for both positive and negative values of k_1. Therefore, we expect that for this system, the maximum steady-state error produced by a distortion that is purely radial in nature would be on the order of the distortion parameter itself.

In Figures 23–25 we see the effects of various values of p_1 and p_2. The error induced by a positive value for p_1 is always of the same sign in steady-state. We have now seen simulations for positive and negative values of distortions which have both symmetric and antisymmetric state estimation error graphs. This provides us with insight into the behavior of the system under

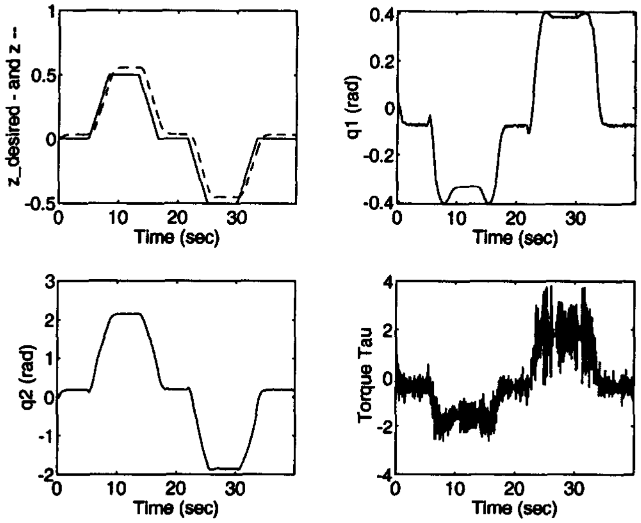

Figure 23. Visual servo with $p_1 = 0.015$ (see text).

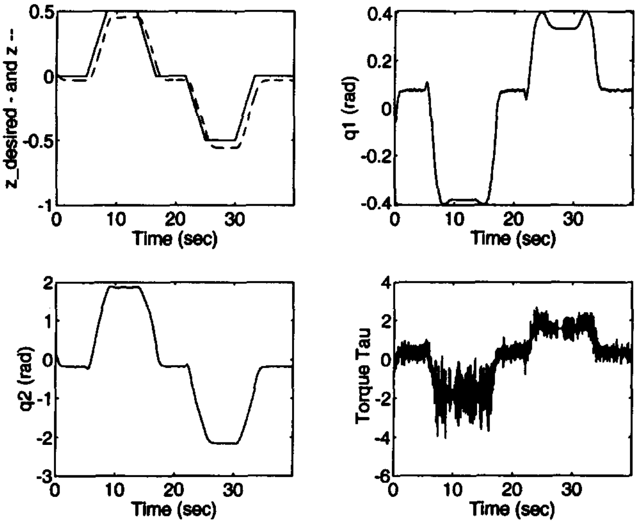

Figure 24. Visual servo with $p_1 = -0.015$ (see text).

various distortions when tracking positive versus negative values of z. Thus, we will henceforth consider only positive z values.

In Figures 26 and 27 we see the effects of thin prism distortions on the closed-loop system. We note that the estimation error graph for s_1 shown in Figure 16 is symmetric with respect to $z = 0$, while that of s_2 in Figure 17 is antisymmetric. Thus, we expect s_1 to produce effects similar to p_1, while s_2 will produce effects similar to p_2. This is borne out by the simulations.

The sizes of the steady-state errors in Figure 26 cannot be easily rationalized from the shape of the estimation error graph for s_1 (Figure 16). Even though the total state estimation error is smaller when the system is in steady-state at large values of z than it is near $z = 0$, the error in z is greatest when the z_{desired} is larger. This behavior is a result of the control algorithm, and should be noted when the system is set up.

5. DISCUSSION

Overall, we see that distortions that contain tangential components are very bad for the performance of this system. This is not necessarily the case for a general system, but is the product

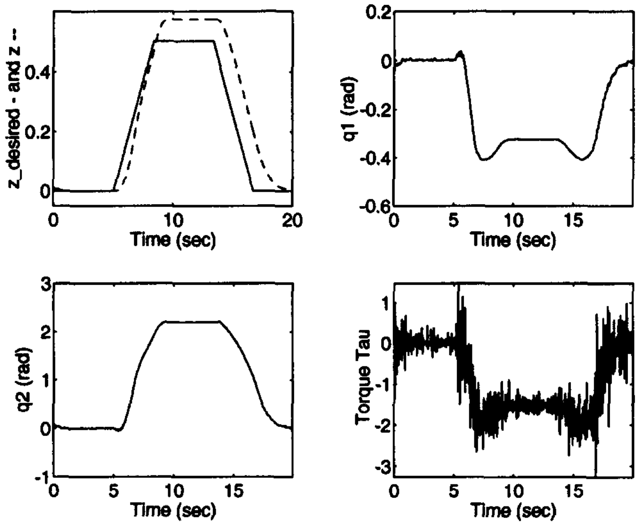

Figure 25. Visual servo with $p_2 = 0.015$ (see text).

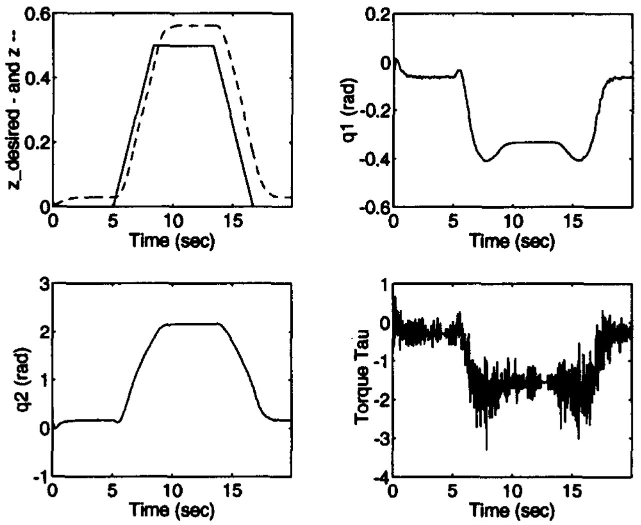

Figure 26. Visual servo with $s_1 = 0.0125$ (see text).

of the special geometry of this system. The proper choice of camera for direct visual servoing of the acrobot would thus be one which has low tangential distortion and a small exposure factor E_t. Further, we note that the acrobot system, given a camera with very low distortions, can be controlled very well using the extremely simple direct visual servo system described in Section 3.2.

As for the stability of the system when distortions are present, we can see that the steady-state error near $z = 0.5$ could easily force the system unstable, were the distortions large enough. We further note that, in the absence of distortions, the system is stable for sample rates as low as 10 Hz, for a slowly varying desired trajectory. Thus, it may be possible to perform better visual servoing of this particular system using more complex methods. For moderate distortions, however, we see that a sample rate of 30 Hz is sufficient. Even so, for a camera with a full sample period exposure time, the sample period should be as low as possible to avoid blurring and to reduce the inherent delay time produced by this algorithm under the effects of motion blur.

A major concern when utilizing a direct visual servo system is the input torque τ. As seen in the simulations of the acrobot under direct visual servo, the control signal has a great deal of

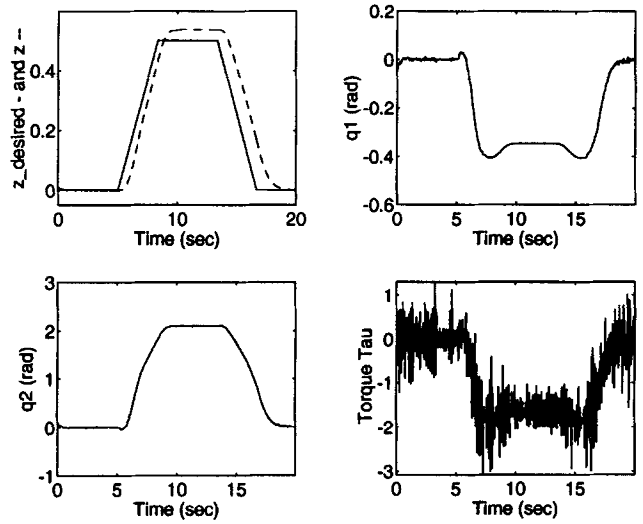

Figure 27. Visual servo with $s_2 = 0.025$ (see text).

chatter, which could present a problem for a real implementation. It is possible that the desired control torque might not be achievable by the actuator, which would have a fixed maximum value of $\dot{\tau}$. This is a difficulty that could lead to additional oscillations in the system, or even to instability.

All simulations pointed toward one additional fact that is not necessarily intuitive. In every case, the steady-state tracking error was greater for a z_{desired} which was large in magnitude than it was when $z_{\text{desired}} = 0$. Thus, for our example, we should arrange the system such that the state estimation error is minimized at the extremes of the unstable equilibrium manifold. This might be accomplished by arranging the robot in the field of view of the camera in a special way based on the existing distortion. This behavior highlights the fact that we should not ignore the specific properties of a controller when designing and implementing a visual servo system.

We have demonstrated the effects of lens distortions and aberrations on the performance of a specific direct visual servo system. A straightforward continuation of this line of endeavor is to consider methods of dealing with such distortions in a general direct visual servo paradigm. A complete discussion of such approaches is beyond the scope of this paper. However, it can be stated that for a well-identified camera system in a relatively constrained visual servo problem, it is possible and perhaps desirable to compensate for some of the distortion.

In the acrobot example, it is possible for us to compensate for lens distortions based on our knowledge of state estimation errors with respect to distortion values, assuming we have calibrated the camera and identified the distortions. This is a simple application of the same simulation used to discuss the estimation errors in Section 4.1.2. In general, each specific problem can offer a variety of methods for compensating for lens distortions.

6. CONCLUSIONS

In this paper, we have presented qualitative and quantitative analyses of the effects of various real-world unmodelled imaging dynamics on visual servo systems, concentrating mainly on the very fundamental issues associated with the simplest form of such control: the direct visual servo system.

We have generated a complete, detailed camera model for use in evaluating various visual algorithms, and have shown an example of a system that is well-suited to direct visual servo control. We have presented an analysis of the estimation errors and stability questions involved with this system and have given concrete examples of the effects of various distortions and disturbances on the system.

We have shown that defocus and image noise have little effect on the type of feature-based data extraction technique utilized herein, but that motion blur may present significant difficulties that require special care to overcome. We have classified a set of lens distortion parameters that quantify the divergence of a real camera lens system from the ideal pinhole lens model. We have seen that the effects of these distortions vary by application and sensing strategy. The analysis of lens distortions has also provided valuable data concerning some general guidelines, that should be followed when there is the possibility of significant lens distortion in a system.

Future work includes the study of the effects of actuator dynamics, such as a fixed $\hat{\tau}$, on the stability of direct visual servo systems in general, and the acrobot case in specific. Inclusion of special processing steps to reduce defocus and motion blur, as well as attempts to generate more accurate velocity calculations and smooth control, are all areas of interest for direct visual servo systems.

REFERENCES

1. M.H. Raibert and J.J. Craig, Hybrid position/force control of manipulators, *ASME Journal of Dynamic Systems, Measurement and Control* **102** (6), 126–133 (1981).
2. T. Yoshikawa, T. Sugie and N. Tanaka, Dynamic hybrid postion/force control of robot manipulators—controller design and experiments, *IEEE Journal of Robotics and Automation* **RA-4** (6), 699–705 (1988).
3. Y. Shirai and H. Inoue, Guiding robot by visual feedback in assembling tasks, *Pattern Recognition* **5**, 99–108 (1973).
4. P. Saraga and B.M. Jones, Simple assembly under visual control, In *Robot Vision*, (Edited by A. Pugh) pp. 209–233, IFS Pub. Ltd., U.K., (1983).
5. P. Allen, B. Yoshimi and A. Timcenko, Real-time visual servoing, In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 851–856, (1991).
6. L.E. Weiss, A.C. Sanderson and C.P. Neuman, Dynamic sensor-based control of robots with visual feedback, *IEEE Journal of Robotics and Automation* **RA-3** (5), 404–417 (1987).
7. J.T. Feddema and O.R. Mitchell, Vision-guided servoing with feature-based trajectory generation, *IEEE Transactions on Robotics and Automation* **5** (5), 691–700 (1989).
8. S.B. Skaar, W.H. Brockman and R. Hanson, Camera-space manipulation, *Interbational Journal of Robotics Research* **6** (4), 20–32 (1987).
9. N. Papanikolopoulos, P.K. Khosla and T. Kanade, Vision and control techniques for robotic visual tracking, In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 857–864, (1991).
10. W.F. Clocksin, J.S.E. Bromley, P.G. Davey, A.R. Vilder and C.G. Morgan, An implementation of model-based visual feedback for robot arc welding of thin sheet steel, *International Journal of Robotics Research* **4** (1), 13–26 (1985).
11. P.K. Khosla, C.P. Neuman and F.B. Prinz, An algorithm for seam tracking applications, *International Journal of Robotics Research* **4** (1), 327–41 (1985).
12. J.E. Agapakis, J.M. Katz, J.M. Friedman and G.N. Epstein, Vision-aided robotic welding: An approach and a flexible implementation, *International Journal of Robotics Research* **9** (5), 17–33 (1990).
13. A. Castano and S.A. Hutchinson, Hybrid vision/position servo sontrol of a robotic manipulator, In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1264–1269, Nice, France, (1992).
14. 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 (1992).
15. K. Hashimoto, T. Kimoto T. Ebine and H. Kinura, Manipulator control with image-based visual servo, In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2267–2272, Sacramento, CA, (1991).
16. A.J. Koivo and N. Houshangi, Real-time vison feedback for servoing robotic manipulator with self-tuning controller, *IEEE Transactions on Systems, Man, and Cybernetics* **21** (1), 134–142 (1991).
17. S.B. Skaar, W.H. Brockman and W.S. Jang, Three-dimensional camera-space manipulation, *International Journal of Robotics Research* **9** (4), 22–39 (1990).
18. O. Khatib, A unified approach for motion and force control of robot manipulators: The operational space formulation, *IEEE Journal of Robotics and Automation* **RA-3**, 43–53 (1987).
19. T.J. Tarn, A. Bejczy, A. Isidori and Y. Chen, Nonlinear feedback in robot arm control, In *IEEE Conf. on Decision and Control*, (1984).
20. P.I. Corke, Dynamics of visual control, In *Proc. IEEE Workshop on Visual Servoing: Achievements, Applications and Open Problems*, (Edited by G. Hager and S. Hutchinson), Inst. of Electrical and Electronics Eng., Inc., (1994).
21. R.Y. Tsai, A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses, *IEEE Journal of Robotics and Automation* **3** (4), 323–344 (1987).

22. J. Weng, P. Cohen and M. Heriou, Calibration of stereo cameras using a nonlinear distortion model, In *Int'l. Conf. on Pattern Recognition*, pp. 246–253, (1990).
23. H.A. Martins, J.R. Birk and R.B. Kelly, Camera models based on data from two calibration plates, *Computer Vision, Graphics, and Image Processing* **17**, 173–180 (1981).
24. G.V. Puskorius and L.A. Feldkamp, Camera calibration methodology based on a linear perspective transformation error model, In *Proceedings of the IEEE International Conference on Robotics and Automation*, 1858–1860, (1980).
25. *Manual of Photogrammetry*, 4th edition, American Society of Photogrammetry, (1980).
26. *CCD Image Sensors and Cameras*, DALSA Inc., Waterloo, Ontario, Canada, (1991).
27. S.A. Bortoff, Pseudolinearizartion using Spline functions with application to the acrobot, Ph.D. Thesis, University of Illinois at Urbana-Champaign, Coordinated Science Laboratory, (1992).
28. C. Reboulet and C. Champetier, Feedback control of nonlinear systems by extended linearization, *International Journal of Control* **40** (4), 631–638 (1984).
29. J. Aloimonos and D.P. Tsakiris, On the mathematics of visual tracking, *Image and Vision Computing* **9** (4), 235–251 (1991).
30. P. Anandan, A computational framework and an algorithm for the measurement of visual motion, *International Journal of Computer Vision* **2**, 283–310 (1989).
31. B.E. Bishop, Direct visual servo control with application to the acrobot, M.S. Thesis, University of Illinois at Urbana-Champaign, Coordinated Science Laboratory, (1994).