CONF-970469--10

IMAGE BASED AUTODOCKING WITHOUT CALIBRATION*

Herry Sutanto
University of Illinois

Rajeev Sharma
Pennsylvania State University

Venugopal K. Varma
Robotics and Process Systems

Oak Ridge National Laboratory†
Post Office Box 2008
Oak Ridge, Tennessee 37831-6304

To be presented at the
International Conference on
Robotics and Automation
Albuquerque, New Mexico

April 1997

*Research sponsored by the U.S. Army's Project Manager, Advanced Field Artillery Systems/Future Armored Resupply Vehicle under Interagency Agreement 1892-A078-A1 between the U.S. Department of Energy and the Armament Research, Development, and Engineering Center at the Picatinny Arsenal.

†Managed by Lockheed Martin Energy Research, Corp., under contract DE-AC05-96OR22464 with the U.S. Department of Energy. .

DISCLAIMER

Image-Based Autodocking Without Calibration

Herry Sutanto
University of Illinois
hsutanto@uiuc.edu

Rajeev Sharma
Pennsylvania State University
rsharma@cse.psu.edu

Venugopal Varma
Oak Ridge National Laboratory
vkv@ornl.gov

Abstract

The calibration requirements for visual servoing can make it difficult to apply in many real-world situations. One approach to image-based visual servoing without calibration is to dynamically estimate the image Jacobian and use it as the basis for control. However, with the normal motion of a robot toward the goal, the estimation of the image Jacobian deteriorates over time. We propose the use of additional "exploratory motion" to considerably improve the estimation of the image Jacobian. We study the role of such exploratory motion in a visual servoing task. Simulations and experiments with a 6-DOF robot are used to verify the practical feasibility of the approach.

1 Introduction

Sensor-based control can play an important role in autonomous robotics. Computer vision, in particular, could help in providing a flexible feedback mechanism for overcoming uncertainties. There has been a growing interest in visual servo control in recent years [1], partly because of a decrease in hardware costs and the advances in computer vision.

Visual servo control, based on the feedback representation mode, can be classified as being either *position based* or *image based*. An image-based servoing system [2, 3] observes how differential changes in robot configuration space relate to differential changes in image features space and then uses this relationship to control the robot motion to achieve the goal. The matrix that captures the relationship between the differential changes in the robot joints and the image features is named the *image Jacobian*. Note that image features here refer to any measurable image parameters that can be used for control, for example, position, size, distance, or surface area of objects in the image.

If the robot and camera are completely calibrated then the image Jacobian can be computed at each robot configuration and becomes a basis for visual servo control. However, the process of *calibration* could be tedious and error prone and in some situations may be infeasible. Thus it is desirable to devise control techniques that can avoid calibration. The image-based approach is appealing because by selecting a right set of image features it may be possible to carry out a task without calibration [4, 5].

In this paper we follow the approach of dynamically estimating the image Jacobian at each step. The estimated image Jacobian then forms the basis of visual control. However, in following this approach, the estimation of the image Jacobian deteriorates since the update is only in the goal direction. To alleviate this situation, we introduce the idea of *exploratory motion* to improve the estimate of the image Jacobian. We consider the issues and trade-offs involved in using the exploratory motions. The study was completed by conducting experiments on a 6-DOF robot and simulations using a computer model of the same robot. The results establish the utility of the exploratory motion for visual servoing without calibration.

2 Servoing Scheme

2.1 The image Jacobian

Assume a robot manipulator with n joints and n degrees of freedom. Assume that a camera is mounted on the end-effector of the robot and that the servoing task is defined in terms of m image features. Let $\mathbf{q} = [q_1, \ldots, q_n]^T$ be the n-dimensional vector that represents a point in the robot configuration space. Let $\mathbf{r} = [r_1, \ldots, r_p]^T$ be the p-dimensional vector that represents the position of the end-effector in a Cartesian coordinate system. Let $\mathbf{f} = [f_1, \ldots, f_m]^T$ be the m-dimensional vector that represents a point in image feature space. The relation between joint velocity of the robot $\dot{\mathbf{q}} = [\dot{q}_1, \ldots, \dot{q}_n]^T$ and its corresponding velocity in task space, $\dot{\mathbf{r}} = [\dot{r}_1, \ldots, \dot{r}_p]^T$, is captured in terms of the robot Jacobian, $\mathbf{J}$, as

$$\dot{\mathbf{r}} = \mathbf{J} \cdot \dot{\mathbf{q}}, \tag{1}$$

where

$$\mathbf{J} = \begin{bmatrix} \frac{\partial r_1}{\partial q_1} & \cdots & \frac{\partial r_1}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial r_p}{\partial q_1} & \cdots & \frac{\partial r_p}{\partial q_n} \end{bmatrix}. \tag{2}$$

A change in the end-effector position results in a change in the image parameters. A perspective projection model can be used to capture this dependence. Thus the feature velocities $\dot{\mathbf{f}} = [\dot{f}_1, \ldots, \dot{f}_m]^T$ are related to the task space velocities as follows:

$$\dot{\mathbf{f}} = \mathbf{J}_\mathbf{r} \cdot \dot{\mathbf{r}} \tag{3}$$

where $\mathbf{J}_\mathbf{r}$ is the $m \times p$ matrix of the local Jacobian at

configuration $\mathbf{q}$:

$$\mathbf{J_r} = \begin{bmatrix} \frac{\partial f_1}{\partial r_1} & \cdots & \frac{\partial f_1}{\partial r_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial r_1} & \cdots & \frac{\partial f_m}{\partial r_p} \end{bmatrix}. \quad (4)$$

Thus the velocity of the image features can be directly related to joint velocities in terms of a composite Jacobian, which we refer to as the *image Jacobian*:

$$\dot{\mathbf{f}} = \mathbf{J_q} \cdot \dot{\mathbf{q}}, \quad (5)$$

where $\mathbf{J_q} = \mathbf{J}.\mathbf{J_r}$, thus

$$\mathbf{J_q} = \begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \cdots & \frac{\partial f_1}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial q_1} & \cdots & \frac{\partial f_m}{\partial q_n} \end{bmatrix}. \quad (6)$$

Since its introduction [6], the image Jacobian has been used in many visual servo control applications [2, 3, 5, 7].

2.2 Dynamic estimation of image Jacobian

The image Jacobian $\mathbf{J_q}$ relates the image feature velocity as a linear combination of joint space velocity at a given robot configuration $\mathbf{q}$. Based on this fact, a simple servoing mechanism can be derived. Because of the discrete nature of the simulation setup, instead of configuration space and feature space velocities, discrete infinitesimal motions in both spaces are assumed in the following description.

Given a goal feature $\mathbf{f}^g$ to be achieved and the currently observed feature vector $\mathbf{f}^c$, a current feature error can be calculated as $\Delta\mathbf{f} = \mathbf{f}^g - \mathbf{f}^c$. Based on the current image Jacobian $\mathbf{J_q}$, a joint motion ($\Delta\mathbf{q}$) that reduces this error is given by

$$\Delta\mathbf{q} = \mathbf{J_q^{-1}}\Delta\mathbf{f}. \quad (7)$$

Since the image Jacobian $\mathbf{J_q}$ is only correct in the neighborhood of $\mathbf{q}$, $\Delta\mathbf{q}$ should be treated as directional information. Hence, the next joint motion, $\mathbf{dq}$, is calculated as a scaled version of normalized $\Delta\mathbf{q}$.

Since no calibration is done, the true image Jacobian at each configuration point is unknown. However, it can be dynamically estimated based on several differential changes in the joint space (joint space displacements) with their corresponding changes in the feature space (feature space displacements).

At step j, the system extracts feature vector $\mathbf{f}^j$ from the image and configuration vector $\mathbf{q}^j$ from the robot joint encoders. Instead of configuration and feature space velocities, the system uses derived configuration space vector displacement $\mathbf{dq}^j = \mathbf{q}^j - \mathbf{q}^{j-1}$ and feature space displacement $\mathbf{df}^j = \mathbf{f}^j - \mathbf{f}^{j-1}$ to estimate the image Jacobian.

In order to uniquely control the robot joints, the feature space dimension must be greater than or equal to the configuration space dimension ($n \leq m$). To get an estimation of image Jacobian $\hat{\mathbf{J}}_\mathbf{q}$, n pairs of configuration and feature space vector displacement $[(\mathbf{dq}^{j-n+1}, \mathbf{df}^{j-n+1}), \ldots, (\mathbf{dq}^j, \mathbf{df}^j)]$ are needed. Matrix $\mathbf{DQ}$ and $\mathbf{DF}$ can be composed by concatenating the above n configuration space displacements and n feature space displacements, respectively.

$$\mathbf{DQ} = \begin{bmatrix} \mathbf{dq}^{j-n+1} & \cdots & \mathbf{dq}^j \end{bmatrix} = \begin{bmatrix} dq_1^{j-n+1} & \cdots & dq_1^j \\ \vdots & \ddots & \vdots \\ dq_n^{j-n+1} & \cdots & dq_n^j \end{bmatrix} \quad (8)$$

$$\mathbf{DF} = \begin{bmatrix} \mathbf{df}^{j-n+1} & \cdots & \mathbf{df}^j \end{bmatrix} = \begin{bmatrix} df_1^{j-n+1} & \cdots & df_1^j \\ \vdots & \ddots & \vdots \\ df_m^{j-n+1} & \cdots & df_m^j \end{bmatrix} \quad (9)$$

Every $\mathbf{dq}^i = [dq_1^i, \ldots, dq_n^i]^T$ and $\mathbf{df}^i = [df_1^i, \ldots, df_m^i]^T$ is an $n-$ and $m-$dimensional column vector respectively; hence, $\mathbf{DQ}$ is an $n \times n$ matrix and $\mathbf{DF}$ is an $m \times n$ matrix. Thus, the estimated image Jacobian is computed as a product matrix given by:

$$\hat{\mathbf{J}}_\mathbf{q} = \mathbf{DF} \cdot \mathbf{DQ}^{-1}. \quad (10)$$

Using the estimated image Jacobian $\hat{\mathbf{J}}_\mathbf{q}$ the next robot motion $\mathbf{dq}^{j+1}$ can be determined based on Equation (7). After the robot executes the prescribed motion, the corresponding feature vector displacement $\mathbf{df}^{j+1}$ is observed. The new vector pair, $\mathbf{dq}^{j+1}$ and $\mathbf{df}^{j+1}$, updates $\mathbf{DQ}$ and $\mathbf{DF}$ matrices for estimating the next cycle of image Jacobian matrix. The latest pair has the most accurate information about the current image Jacobian; hence, the easiest update can be done by replacing the oldest pair $(\mathbf{dq}^{j-n+1}, \mathbf{df}^{j-n+1})$ with the newly acquired pair $(\mathbf{dq}^{j+1}, \mathbf{df}^{j+1})$.

2.3 Initial estimate of the image Jacobian

When the visual servoing starts, $\mathbf{DQ}$ and $\mathbf{DF}$ can be acquired either (1) by random movements near the initial configuration or (2) by systematically making small moves one joint at a time and observing the corresponding feature vector displacements. The latter approach is preferred because it guarantees capturing all possible robot joint movements, thus guaranteeing that $\mathbf{DQ}$ is well behaved and its inverse can be calculated accurately. This process is called *acquiring the true image Jacobian* in contrast to *estimating the image Jacobian*, which is done at every step toward achieving the goal as given by Equation (10).

3 Exploratory Motion

Image-based servoing estimates the current image Jacobian based on the observed relationship between joint movement and image feature changes. The estimation should consider only those relationships near the current configuration; otherwise, the estimation would be less accurate. We consider how the estimation could be related to the normal goal-seeking motion or a more deliberate motion aimed at improving the estimation of the image Jacobian.

(i) (ii) (iii)

Figure 5: Camera view of the port as the visual servoing progresses.

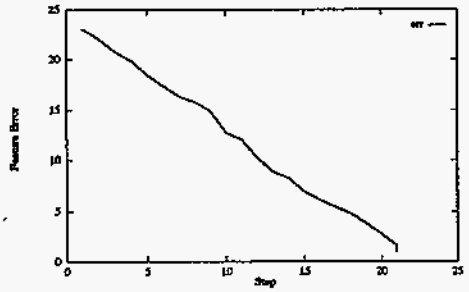

Figure 6: Feature error profile during servoing.

Thus the set of experiments helped in establishing the feasibility of an image-based servoing scheme without calibration and showed its flexibility in terms of arbitrary camera positions and port locations. The next subsection considers simulation experiments to study the effect of the exploratory motion on the performance of the servoing.

4.3 Simulations to study exploratory motion

The simulations were conducted using the model of the 6-DOF Mitsubishi robot described earlier. The simulations were used to study the relative benefits and costs involved in using the exploratory motions.

In this study, around 200 cases were simulated. For each case, an initial configuration within the effective range of the robot was generated randomly. A servoing algorithm with exploratory motion was used to dock the robot. Several behaviors of this docking with exploratory motion were recorded. Henceforth, this method is called EXPL for ease of reference. The same problem was presented to the basic servoing algorithm without the exploratory motion, referred to as BSC, and then the two results were compared and analyzed. For benchmarking purposes, the same problem was also presented to an algorithm with complete information of the image Jacobian; i.e., the image Jacobian is acquired at every step. We refer to this method as INF.

Three different measurements are used to evaluate the performance of these methods: (1) the number of steps to converge to the goal, (2) the average directional error, and (3) the average condition number of the estimated image Jacobian.

Number of steps to converge. Table 1 presents the statistics of the mean and standard deviation of the number of steps to converge for each method. It shows that adding exploratory motions (EXPL) on the average reduces the number of steps needed to converge (from 72.317 steps to 59.374 steps). The improvement is significant since it is close to the best possible performance of 57.268 steps with the expensive method of reacquiring the image Jacobian at every step (INF).

	Number of steps to converge		
	BSC	EXPL	INF
Mean	72.317	59.374	57.268
Std. Dev.	37.259	27.584	29.108

Table 1: The number of steps to converge for each method.

Directional error. Another way to measure the performance of both the basic and exploratory methods is to measure how close the approximated image Jacobian is to the actual image Jacobian. This can be calculated as the directional error of the gradient of the approximated image Jacobian from the actual correct gradient to the goal. In other words, given the current feature error $\Delta\mathbf{f}$, the approximated image Jacobian $\hat{\mathbf{J}}_\mathbf{q}$ prescribes a direction $\Delta\hat{\mathbf{q}} = \hat{\mathbf{J}}_\mathbf{q} \cdot \Delta\mathbf{f}$ in configuration space. If the actual image Jacobian at current configuration point $\mathbf{J}_\mathbf{q}$ is known, the "correct" direction in configuration space can be calculated as $\Delta\mathbf{q} = \mathbf{J}_\mathbf{q} \cdot \Delta\mathbf{f}$. The angle between directional information given by $\Delta\hat{\mathbf{q}}$ and $\Delta\mathbf{q}$ is considered as the *directional error*.

Table 2 shows that adding exploratory motions on the average reduces the directional errors (from 52.42° to 50.24°). As a side note, in general a directional error less than 90° implies that the prescribed direction and the best direction are at the same n-dimensional hemisphere; hence, it usually still reduces the feature error even though it is not at the greatest descent.

	Directional Errors	
	BSC	EXPL
Mean	52.42°	50.24°
Std. Dev.	18.99°	21.84°

Table 2: Statistics of directional errors.

Condition number of matrix MQ. As mentioned in Section 3.4, matrix MQ represents the configuration space ellipsoid which defines the level of confidence of knowing the correlation of joint movement in that direction with its corresponding feature space changes.

Hence, for the INF method where at every step a correct image Jacobian is acquired by moving a single joint at a time, this confidence ellipsoid has a perfect spherical shape (the feature changes in all directions of joint space are equally known). Thus, the condition number is always 1 or its logarithmic value is 0. The smaller the condition number, the closer the confidence ellipsoid to a sphere.

The statistics in Table 3 show that having exploratory motion improves the shapes of the confidence ellipsoid, that is, a more balanced information in all directions in the joint space.

4.4 Conservative vs aggressive exploration

All of the above results are gathered with maximum weight of exploratory motion set to be one-half, or, in

Jacobian estimation and the cost in terms of the path

(i) (ii) (iii)

Figure 5: Camera view of the port as the visual servoing progresses.

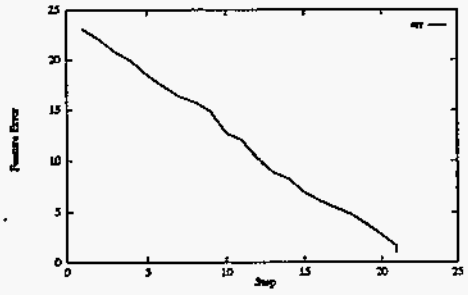

Figure 6: Feature error profile during servoing.

Thus the set of experiments helped in establishing the feasibility of an image-based servoing scheme without calibration and showed its flexibility in terms of arbitrary camera positions and port locations. The next subsection considers simulation experiments to study the effect of the exploratory motion on the performance of the servoing.

4.3 Simulations to study exploratory motion

The simulations were conducted using the model of the 6-DOF Mitsubishi robot described earlier. The simulations were used to study the relative benefits and costs involved in using the exploratory motions.

In this study, around 200 cases were simulated. For each case, an initial configuration within the effective range of the robot was generated randomly. A servoing algorithm with exploratory motion was used to dock the robot. Several behaviors of this docking with exploratory motion were recorded. Henceforth, this method is called EXPL for ease of reference. The same problem was presented to the basic servoing algorithm without the exploratory motion, referred to as BSC, and then the two results were compared and analyzed. For benchmarking purposes, the same problem was also presented to an algorithm with complete information of the image Jacobian; i.e., the image Jacobian is acquired at every step. We refer to this method as INF.

Three different measurements are used to evaluate the performance of these methods: (1) the number of steps to converge to the goal, (2) the average directional error, and (3) the average condition number of the estimated image Jacobian.

Number of steps to converge. Table 1 presents the statistics of the mean and standard deviation of the number of steps to converge for each method. It shows that adding exploratory motions (EXPL) on the average reduces the number of steps needed to converge (from 72.317 steps to 59.374 steps). The improvement is significant since it is close to the best possible performance of 57.268 steps with the expensive method of reacquiring the image Jacobian at every step (INF).

	Number of steps to converge		
	BSC	EXPL	INF
Mean	72.317	59.374	57.268
Std. Dev.	37.259	27.584	29.108

Table 1: The number of steps to converge for each method.

Directional error. Another way to measure the performance of both the basic and exploratory methods is to measure how close the approximated image Jacobian is to the actual image Jacobian. This can be calculated as the directional error of the gradient of the approximated image Jacobian from the actual correct gradient to the goal. In other words, given the current feature error $\Delta\mathbf{f}$, the approximated image Jacobian $\hat{\mathbf{J}}_\mathbf{q}$ prescribes a direction $\Delta\hat{\mathbf{q}} = \hat{\mathbf{J}}_\mathbf{q} \cdot \Delta\mathbf{f}$ in configuration space. If the actual image Jacobian at current configuration point $\mathbf{J}_\mathbf{q}$ is known, the "correct" direction in configuration space can be calculated as $\Delta\mathbf{q} = \mathbf{J}_\mathbf{q} \cdot \Delta\mathbf{f}$. The angle between directional information given by $\Delta\hat{\mathbf{q}}$ and $\Delta\mathbf{q}$ is considered as the *directional error*.

Table 2 shows that adding exploratory motions on the average reduces the directional errors (from 52.42° to 50.24°). As a side note, in general a directional error less than 90° implies that the prescribed direction and the best direction are at the same n-dimensional hemisphere; hence, it usually still reduces the feature error even though it is not at the greatest descent.

	Directional Errors	
	BSC	EXPL
Mean	52.42°	50.24°
Std. Dev.	18.99°	21.84°

Table 2: Statistics of directional errors.

Condition number of matrix MQ. As mentioned in Section 3.4, matrix **MQ** represents the configuration space ellipsoid which defines the level of confidence of knowing the correlation of joint movement in that direction with its corresponding feature space changes.

Hence, for the INF method where at every step a correct image Jacobian is acquired by moving a single joint at a time, this confidence ellipsoid has a perfect spherical shape (the feature changes in all directions of joint space are equally known). Thus, the condition number is always 1 or its logarithmic value is 0. The smaller the condition number, the closer the confidence ellipsoid to a sphere.

The statistics in Table 3 show that having exploratory motion improves the shapes of the confidence ellipsoid, that is, a more balanced information in all directions in the joint space.

4.4 Conservative vs aggressive exploration

All of the above results are gathered with maximum weight of exploratory motion set to be one-half, or, in

	Average $\log_{10}$ *of the Condition Numbers of* MQ	
	BSC	EXPL
Mean	9.133	6.592
Std. Dev.	1.826	2.150

Table 3: The log average of the condition number of MQ.

other words, the minimum portion of the goal reaching motion to be retained [variable p of Equation (16)] is 0.5. A more aggressive exploratory motion can be attained by setting p to a lower value. Thus, when the feature error is high, exploratory motion can exceed the goal-reaching motion. Alternatively, we can set more conservative exploratory motion by raising p. Our experiments show that, for our cases, setting p to 0.6 yields the best average result.

Another way of varying the aggressiveness of exploration is by adjusting the condition number threshold of matrix MQ. In all of the above simulations, exploratory motion is added every time the condition number of MQ passes 100 (or 2 in $\log_{10}$ scale). The higher the condition number threshold, the more conservative the exploration. The effect of varying the condition number threshold is shown in Figure 7. This figure also shows that setting the log of condition number threshold to around 2.5 results in the best average performance.

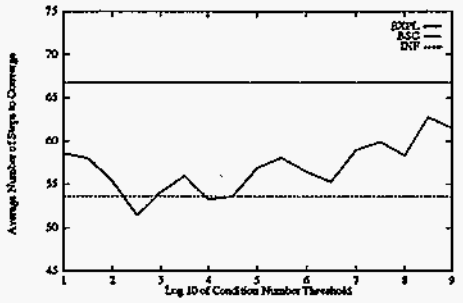

Figure 7: The effect of varying the condition number threshold on the EXPL method.

Even though the particular numbers presented here will not be the same for different robots or different tasks, the experiment shows that there is an optimum mixture of goal reaching movement and exploratory movement which produces the best performance: an overly conservative exploratory method will still suffer the same problem as the basic approach (incorrect Jacobian), whereas an overly aggressive exploratory method may end up exploring the space without reaching the goal.

5 Discussion and Conclusions

The context of the exploratory motion is the dynamic estimation of the image Jacobian. Other calibration-free approaches could also have been used. Alternatively, the method could have been combined with approaches that use partial calibration. The study helps in understanding the interplay between the improvement in the image Jacobian estimation and the cost in terms of the path length. A basis is provided for introducing the requisite magnitude and direction of the exploratory motion to meet the needs of the servoing task.

One disadvantage of the image-based approach is that the effective working space is small since the goal needs to be visible to the camera at all stages of the servoing task. When features *black out* occurs during servoing, a simple strategy of using the last state with all features visible can be employed, thereby enlarging the working space. However, even with the limited workspace, some interesting problems can be solved, for example, the autodocking problem for the Army resupply vehicle which was the task that originally motivated the study.

The performance of an image-based autodocking system highly depends upon the set of features chosen for controlling the robot. Research has been devoted to the issues of feature selection, e.g., [8, 9]. Most of these measures relate to how the features can monitor errors as they occur; hence, a better control system can be derived. However, some features may be better for positioning, while others are better for orientation control. Future work needs to explore a performance measure that considers this decoupling.

Acknowledgment. *We would like to thank B. Jatko, J. Goddard and M. Zeller for their help in conducting the experiments. Research sponsored in part by the U.S. Army's Project Manager Crusader under Interagency Agreement 1892-A078-A1 between the U.S. Department of Energy and the Armament Research, Development, and Engineering Center at the Picatinny Arsenal.*

References

[1] P. I. Corke. Visual control of robot manipulators—a review. In K. Hashimoto, editor, *Visual Servoing*, pp. 1–32. World Scientific, 1993.

[2] J. T. Feddema and O. R. Mitchell. Vision-guided servoing with feature-based trajectory generation. *IEEE Trans. Robotics and Automation*, 5:691–700, 1989.

[3] K. Hashimoto, T. Kimoto, T. Ebine, and H. Kimura. Manipulator control with image-based visual servo. In *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 2267–2271, 1991.

[4] G. D. Hager. Real-time feature tracking and projective invariance as a basis for hand-eye coordination. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 533–539, 1994.

[5] B. Yoshimi and P. K. Allen. Active, uncalibrated visual servoing. In *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 156–161, San Diego, CA, May 1994.

[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*, 3:404–417, 1987.

[7] N. P. Papanikolopoulos, P. K. Khosla, and T. Kanade. Visual tracking of a moving target by a camera mounted on a robot: A combination of vision and control. *IEEE Trans. Robotics and Automation*, 9(1):14–35, 1993.

[8] H. Sutanto and R. Sharma. Global performance evaluation of image features for visual servo control. *Journal of Robotic Systems*, 13(4):243–256, 1996.

[9] R. Sharma and S. Hutchinson. Optimizing hand/eye configuration for visual-servo systems. In *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 172–177, May 1995.