Proceedings of the 2002 IEEE
International Symposium on Intelligent Control
Vancouver, Canada • October 27-30, 2002

A Switching Approach to Visual Servo Control

Nicholas R. Gans and Seth A. Hutchinson

ngans@uiuc.edu, seth@uiuc.edu
Dept. of Electrical and Computer Engineering
The Beckman Institute for Advanced Science and Technology
University of Illinois Urbana Champaign
Urbana, IL, USA

Abstract—*In the recent past, many researchers have developed control algorithms for visual servo applications. In this paper, we introduce a new switching approach, in which a high-level decision maker determines which of two low-level visual servo controllers should be used at each control cycle. We introduce two new low-level controllers, one that relies on the homography between initial and goal images, and one that uses an affine transformation to approximate the motion between initial and goal camera configurations. Since an affine transformation can only approximate a restricted set of camera motions, this choice of low-level controllers illustrates the strength of the switching approach. We have evaluated our approach with several simulations, using three candidate switching rules. Although our results are preliminary, we believe that the proposed method is very promising for visual servo tasks in which there is a significant distance between the initial and goal configurations.*

I. Introduction

The dawn of image based visual servo (IBVS) control research came in the mid-eighties, with the introduction of the image Jacobian matrix [1, 2]. IBVS proved to be computationally efficient as well as robust with respect to to calibration and signal errors, ideal for tasks involving interaction with a dynamic environment such as object tracking or positioning a robot tool in an uncertain environment.

For the next several years, researchers proposed a variety of image-based control systems that relied either explicitly or implicitly on the linearization embodied by the image Jacobian (e.g., [3, 4]). During this time, researchers were mostly content to provide local results, and experiments typically involved only small motions of the camera (or robot end effector). More than a decade later Chaumette outlined a number of problems that cannot be solved using the traditional local linearized approaches to visual servo control [5]. This sparked a new wave of research aimed at finding partitioned control schemes in which certain degrees of freedom are controlled using the image Jacobian linearization while others are controlled using the computer vision system to reconstruct certain aspects of the 3D world [6–9].

We recently analyzed the performance of several partitioned visual servo controllers under a variety of conditions [10]. As would be expected, we have found that each of these controllers has its own strengths and weaknesses, and that these are dependent upon many factors, including the amount noise in the imaging system, the task specification, and the methods available for estimating the depth to the target.

Motivated by these observations, we have begun to investigate approaches based on the theory of hybrid dynamical systems, i.e., systems that comprise a set of continuous subsystems along with a rule that switches between them [11, 12] The goal of our research is to develop a system that uses a supervisory controller to select at each control cycle the most appropriate controller from a suite of visual servo controllers. Thus, we are particularly interested in the specific topic of switched control systems. This paper represents our first steps in this direction. We have developed a control system that uses two visual servo controllers, and we have investigated several switching rules to arbitrate between them. The two controllers do not rely on the image Jacobian, and one of them is applicable only for a limited set of tasks; this further illustrates the strength of the switching approach. We believe that our results, though quite preliminary, indicate that our approach is promising.

The remainder of the paper is organized as follows. In Section II we provide a very brief introduction to switched control systems as we use them in our approach. Section III introduces the two visual servo controllers we have selected for our preliminary investigations. Section IV presents the methods that we use to switch between these controllers. The performance of the system are presented in V. Finally, possible directions of future research are discussed in Section VI.

II. Switched Systems

The theory of hybrid dynamical systems, i.e., systems that comprise a number of continuous subsystems and a high-level decision maker that switches between them, has received considerable attention in the control theory community. In this paper, we are concerned with the specific topic of switched control systems. In particular, we present a controller that uses a high level decision maker to select from two visual servo controllers.

In general, a hybrid dynamical system can be represented by the differential equation

$$\dot{x} = f_\sigma(x) : \sigma \in \{1..n\} \tag{1}$$

where f_σ is a collection of n distinct functions. For our purposes, it is convenient to explicitly note that the switching behavior directly affects the choice of the control input u

$$\dot{x} = f(x, u_\sigma) : \sigma \in \{1, 2\}. \tag{2}$$

In our system, each visual servo controller provides a velocity screw, $u = [T_x, T_y, T_z, \omega_x, \omega_y, \omega_z]^T$, and a switching rule determines which is used as the actual control input at each control cycle.

The stability of a switched system is not insured by the stability of the individual controllers. Indeed, a collection of stable systems can become unstable when inappropriately switched, and unstable systems can be made asymptotically stable through switching. To insure stability, it must be proved that all switches

0-7803-7620-X/02/$17.00 © 2002 IEEE

from one system to another cannot result in an unstable output, regardless of the time of the switch or the state of the system.

Lyapunov stability of switched systems can be proven under certain conditions [13, 14]. Generally this requires establishing a common Lyapunov function which works for all systems. Alternately, one can establish a family of Lyapunov functions for the systems such that at each switch, the value of the function at the end of that interval is less than the value of the function of the interval that proceeded it.

In our initial investigations reported here, we have not yet formally proven the stability of our system. We have, however, performed extensive empirical evaluations that demonstrate the efficacy of our approach, and some of these are presented below.

III. Two Visual Servo Controllers

At present, we have implemented two controllers (the minimum required to investigate switching behaviors). The first is related to controllers developed in [6,8], and relies on computing a homography between the initial and goal images. The second approximates the relationship between initial and goal images by an affine transformation. The trade-offs and the rule used to switch between the two controllers are discussed below in Section IV. Here we briefly describe the two controllers.

We denote image feature points by $\mathbf{f}$, feature points in a homogeneous coordinate system by $\tilde{\mathbf{f}}$ and any feature or parameter in the goal image will be designated with an asterisk(*).

A. A Homography-Based Controller

The homography method exploits the epipolar constraints between two images of planar feature points. The homography matrix has been used in previously for visual servoing in [6, 8] to control a restricted set of degrees of freedom. We however, use it to generate a control for all degrees of freedom.

Define $\tilde{\mathbf{f}}^*$, $\tilde{\mathbf{f}}$, as the homogeneous coordinates in two images of a set of 3D points lying on a plane π. These are related by

$$\tilde{\mathbf{f}}^* = \mathbf{H}\tilde{\mathbf{f}} \tag{3}$$

where $\mathbf{H}$ is the calibrated homography matrix. As shown in [15, 16], $\mathbf{H}$ can be decomposed as

$$\mathbf{H} = \mathbf{R}(\mathbf{I}_3 - \frac{\mathbf{t}\mathbf{n}^T}{d}) \tag{4}$$

where $\mathbf{I}_3$ is a 3×3 identity matrix and $\mathbf{R}$ and $\mathbf{t}$ are the rotation matrix and translation vector, respectively, relating the two camera views. The parameter $\mathbf{n}$ is the the normal of the plane π and describes the orientation of π with respect to the current camera view; d is the distance from the current camera origin to the plane π. We calculate the vector $\mathbf{T} = [T_x, T_y, T_z]^T = \hat{d}\mathbf{t}$, where $\hat{d}$ is an estimate of d. From the rotation matrix $\mathbf{R}$, we extract the roll, pitch and yaw angles, which we use as $\omega_x, \omega_y, \omega_z$ to obtain the velocity screw $u = k[T_x, T_y, T_z, \omega_x, \omega_y, \omega_z]$ in which k is a scalar gain constant, or a 6×6 gain matrix..

B. Affine-Approximation Controller

For camera motions that do not involve rotation about the camera x- or y- axes, the initial and goal images will be related by an affine transformation. While this is a constrained set of motions, it is common in many situations, such as aligning camera with a component on a conveyer belt.

Define $\mathbf{f}^*$, $\mathbf{f}$, as the calibrated pixel coordinates of two points in the image plane. Then these points are related by the affine transformation

$$\begin{aligned} \mathbf{f}^* &= \mathbf{A}\mathbf{f} + \mathbf{b} \\ &= \begin{bmatrix} 1 & a \\ 0 & 1 \end{bmatrix} \begin{bmatrix} s_1 & 0 \\ 0 & s_2 \end{bmatrix} \begin{bmatrix} C_\theta & -S_\theta \\ S_\theta & C_\theta \end{bmatrix} \begin{bmatrix} f_x \\ f_y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \end{aligned} \tag{5}$$

in which C_θ and S_θ denote respectively $\cos\theta$ and $\sin\theta$; f_x and f_y are image point coordinates; a, s_i, and θ describe the skew, scale, and rotation respectively; and $\mathbf{b}$ is the translation. Both $\mathbf{A}$ and $\mathbf{b}$ can be obtained by solving a linear system of equations, and QR decomposition can then be used to determine a, s_i, and θ.

Given a, s_i, θ, t_x, t_y we again have the position and orientation relating the initial and goal camera positions. This controller provides the velocity screw $u = k[t_x, t_y, s_2, 0, 0, \theta]$ where k is a gain constant or matrix. Note that if there is no rotation about the x- or y-axes, we will have $s_1 = s_2$.

In general, two images are unlikely to be perfectly related by an affine transformation. However, in the absence of signal noise the approximation is often fairly accurate. In our scheme, it is left to the switching rule to determine when it is appropriate to use this controller.

IV. Methodology of System Switching

To date, we have explored three switching rules. The first is deterministic, and is based on our evaluation of the performance of each controller under various conditions. The other two switching rules include a nondeterministic element.

A. Comparison of the Two Controllers

The major strength of the homography-based controller in our system is that it is the only controller capable of handling general motions which include rotation about the x- and y-axes. If the camera motion does not involve such rotations, the two approaches have similar performance. However, in the presence of noise, the affine method is much more accurate. We conducted a series of Monte Carlo tests in which both systems performed an identical affine motion under the effects of increasing white noise. The homography-based method typically had an error in the pose estimation that was fifty times greater than the affine approach, and error in the total rotation was almost fifteen times greater.

With regard to speed, the affine approximation controller requires far fewer calculations per iteration. However, both systems are capable of operating at above the camera frame rate of 30Hz..

B. Introduction of Switching Rules

In this section we describe the three switching rules.

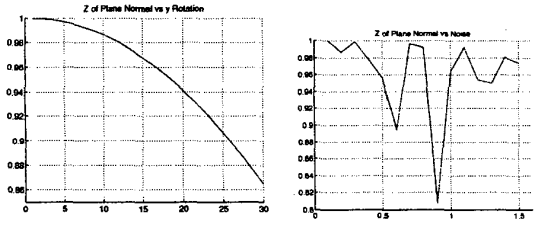
Deterministic Switching. The degree to which a camera motion can be approximated by an affine transformation can be determined by examining the normal $\mathbf{n}$ of the plane π that contains the 3D feature points. In particular, if the motion is in fact affine, then we will have $\mathbf{n} = [n_x, n_y, n_z]^T = [0, 0, 1]^T$. Our

first switching rule is to use the affine-approximation controller when the value of n_z falls below a predetermined threshold.

The threshold value is arbitrary, a lower value will force the system to choose the affine method for larger rotations about the x- and y-axes. In general, the affine method can zero the error for any motion with a rotation about y or x less then 5°, and Figure 1(a) shows that the value of n_z is related to the amount of rotation through an inverse exponential relationship. Signal noise adds a random offset to n_z, and as seen in Figure 1(b), Monte Carlo simulations could discern no clear dependence upon increasing noise variance.

Random Switching. The previous switching rule captures the full abilities of the homographic method, but fails to fully utilize the benefits of the affine system in the case of noise. Our second switching rule is to randomly select, with equal likelihood, one controller at each iteration. Random switching has been used in network control systems to handle task routing [17]. A random choice is the simplest decision mechanism, and in a stable hybrid system will result in an averaging of the performance of all systems.

Biased Random Switching. Finally, a compromise can be struck between the two switching methods. At each iteration the affine approximation controller is selected with a probability that is a monotonic function of n_z, since, as noted earlier, a smaller value for n_z indicates that the homographic method is more suitable. In fact, n_z itself provides a suitable probability to select the affine approximation controller. However, as seen in Figure 1(a), n_z decreases very little even for large rotations about the x- or y-axes. Therefore, we have used an exponential function of n_z to define the probability of selecting the affine approximation controller. Thus, it is increasingly likely to choose the homographic system as n_z decreases, but the random element still allows the system to use the strengths of both systems in the face of random errors.



(a) (b)

Fig. 1. n_z vs. (a) Degrees of y Rotations, (b)Noise Variance

V. Results

We present the results of several simulated tests of the hybrid system, using all switching methodologies described in Section IV. We performed tests for a variety of motion tasks and noise levels. For the deterministic switching, we tested a range of threshold values. Simulations were performed for an ideal camera with a 512 × 512 pixel array, with each pixel measuring $10\mu m \times 10\mu m$ and a focal length of 7.8mm. Depth was calculated as the absolute depth, at each iteration, to the feature point plane, a value known in our simulations. Visual servoing was halted if the pixel error was reduced below 1 pixel, or had converged to steady state for ten iterations.

A. Deterministic Switching

Figure 2 shows results for an affine motion involving camera translation along x, y and z, and rotation about the optical axis with the switching threshold set to 0.99. The top left image shows feature point trajectory during visual servoing. Black portions are motions induced by the affine method, while cyan portions are induced by the homographic method. The majority of this motion was performed by the affine method, however the system did switch to the homographic method at the very end, indicating that for very small motions the system has trouble accurately decomposing the homography matrix. However, very little motion remained be performed, so this switch had no adverse effects. One solution to this non-ideal switch would be to include a weighting factor based on the distance to goal.

The upper right graph shows the pixel error for the four feature points; the pixel error is zeroed in just over fifty iterations. The cyan vertical line indicates a switch to the homographic method after forty iterations, however almost no real motion was performed after this point. The lower two graphs show the translation in meters and rotations in degrees, again a cyan line indicates a transition to the homographic method. For the results in Figure 3 a rotation of thirty degrees of the feature point plane about both the world z- and y-axes is followed by the same affine motion as for the previous test. This causes a general motion involving all degrees of freedom. Again, the top left image shows the feature point trajectory; here we see the homographic method used for the first portion of the motion as indicated by the cyan lines, with a switch to the affine method when the x and y rotation have been reduced. The remaining graphs again show the feature point error and velocities. The vertical black line shows that the switch to the affine method occurred at about the twelfth iteration, and causes a slight incongruity in the velocities, though the error remains fairly smooth. For the simulations illustrated in Figure 4, we used the same affine motion as for

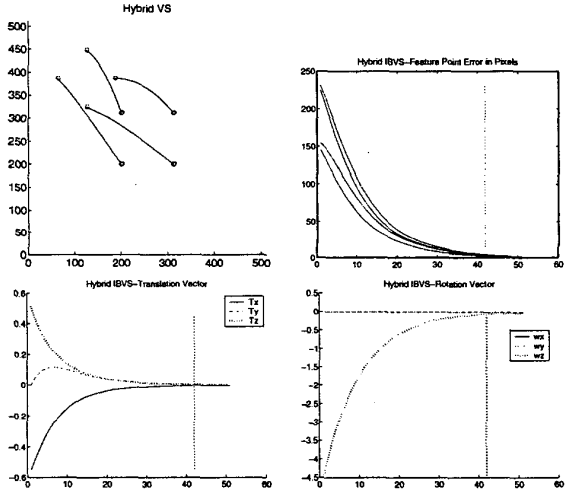


Fig. 2. Pure Affine Motion, Deterministic Switching

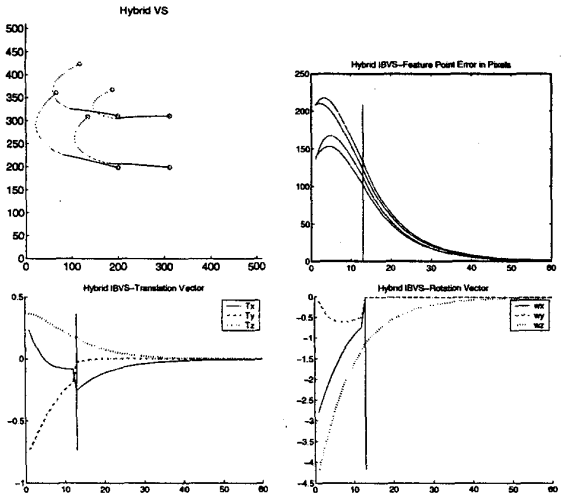


Fig. 3. General Motion, Deterministic Switching

the first test, but we perturbed the locations of the image feature points with additive Gaussian white noise with a variance of 0.5 pixels. There are several instances of the system switching to the homographic method as noise forces the calculated n_z below the threshold value of 0.99. Generally this switch is very brief, and the general motion is quite similar to the case of Figure 2. The velocities are sometimes discontinuous when the system is switched, but the error still converges to zero smoothly, and it actually converges a bit faster than in the noise free case. The

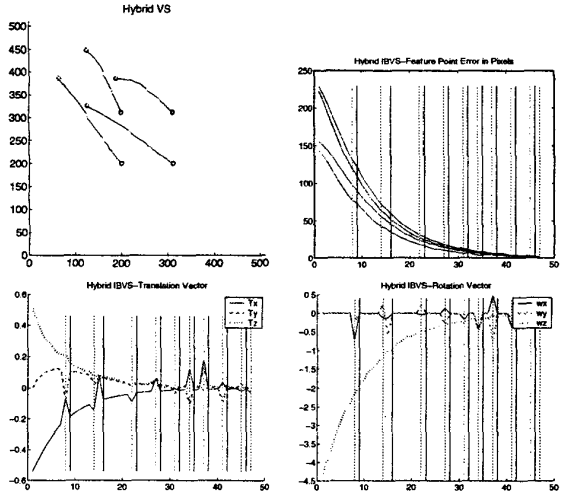


Fig. 4. Affine Motion with Noise, Deterministic Switching

results illustrated in Figure 5 are for the same general motion as the results of Figure 3, but we again perturbed the locations of the image feature points with additive Gaussian white noise with a variance of 0.5 pixels. The system remains to the homographic method for almost all of the motion, briefly switching to the affine method at several points when the noise causes the n_z term to exceed the threshold. The system still zeros the error, though it takes longer than previous tests, over 75 iterations. The velocities are extremely rough in appearance due to the noise. To illustrate the effect of the switching threshold, we

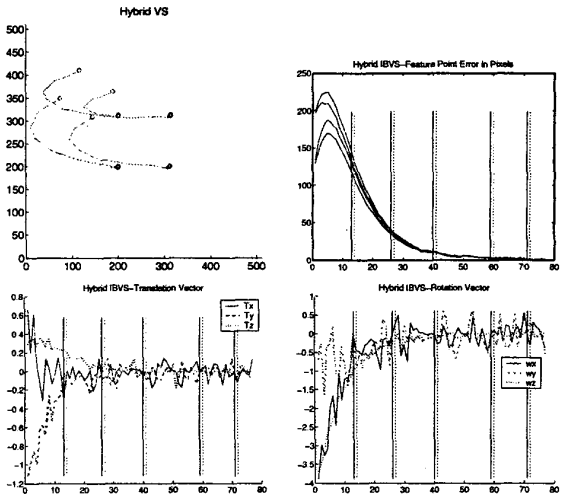


Fig. 5. General Motion with Noise, Deterministic Switching

ran the previous test with a threshold value for n_z of 0.95; results are shown in Figure 6. The trajectory appears much more like the noise free case, with initial homography motions, followed by the affine method, predominantly. The system does take noticeably longer to converge, due to noise effects and possibly remaining necessary rotations about x and y.

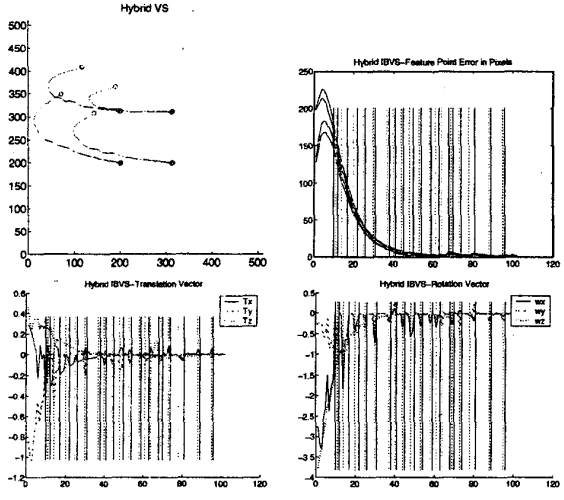


Fig. 6. General Motion with Noise, Deterministic Switching with Lower Switch Threshold

B. Random Switching

Figure 7 shows results for motion involving camera translation along x-, y- and z-axes, and rotation about the optical axis. One of the two controllers is selected at random at each iteration. Again, the top left image shows feature point trajectories, with

black lines indicating use of the affine-approximation controller and cyan portions indicating the homography-based controller. The remaining plots show the feature point error, translation in centimeters, and rotations in degrees. The feature point error is zero after a little more then fifty iterations. Again, black and cyan lines indicate a switch to the homographic and affine methods respectively. Figure 8 shows test results for a general motion involving all degrees of freedom. The trajectory and motion vectors are generally rather rough, though the feature point errors are rather smooth and converge at about the same rate as the switching method. This disjointed appearance is due to the fact that the homographic system induces rotational motions which the affine motion cannot. Notice that as the x and y rotation velocities approach zero, the trajectory becomes fairly smooth as both systems generate similar trajectories for the remaining translations and optical axis rotation. It is worth noting that this system avoids the large feature point motions of the deterministic switching system, which almost loses the feature points from the image plane. Figure 9 shows results for a pure affine motion with random switching between methods, while under degradation of Gaussian white noise with variance of 0.5. The system performs quite similarly to the noise free case. This is expected since, unlike the deterministic switch system of V.1 the noise cannot introduce a random element to the switching. The system does take slightly longer to converge in this test. The final test in this series is the general motion along with added Gaussian noise; results are shown in Figure 10. The velocities again look similar to the noise free case with with large jumps in motion accompanying the switches in systems. The system again takes slightly longer to converge to zero error, and the plot of errors looks noticeably rougher as the noise degrades the ability to compute the correct transformations. The error is successfully reduced below one pixel.

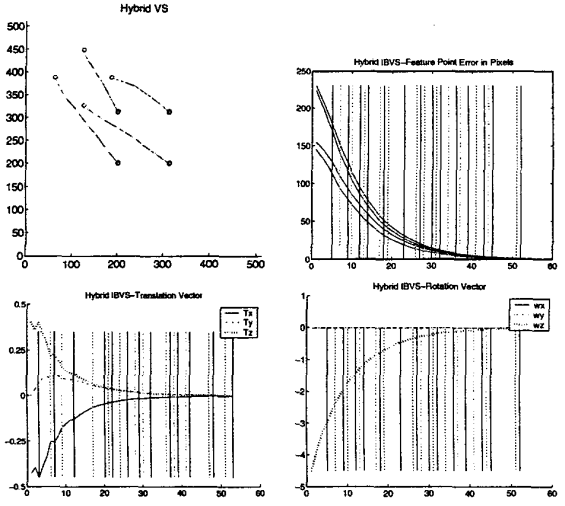


Fig. 7. Pure Affine Motion, Random Switching

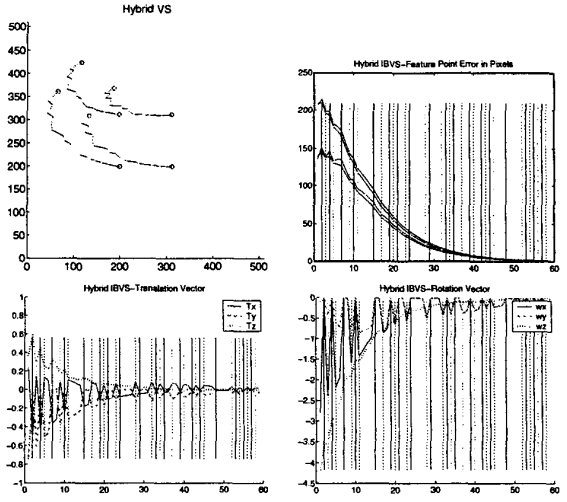


Fig. 8. General Motion, Random Switching

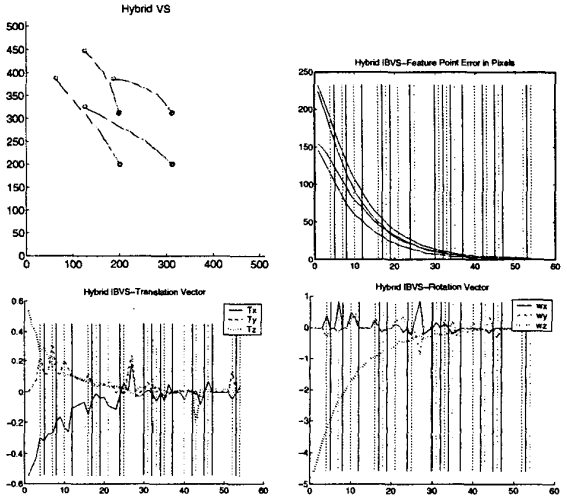


Fig. 9. Pure Affine Motion with Noise, Random Switching

C. *Biased Random Switching*

With this switching rule, at each iteration the decision maker chooses the affine method with probability $\rho = (n_z)^5$. This function keeps the probability of selecting the affine method high when n_z is approximately one, but quickly becomes more likely to select the homography-based method even for slight rotations about the x- and y-axes.

Figure 11 shows the results of the test during pure affine motion. Like the deterministic switching system, the biased random switching approach uses the affine method for almost the entire range of motion. It zeroes the error in approximately fifty iterations. The performance of this switching rule for a general motion is shown in Figure 12. It again has a similar appearance to the deterministic switching system, remaining in the homography-based method for the first portion, while performing x and y rotations. At this point the probabilistic effects

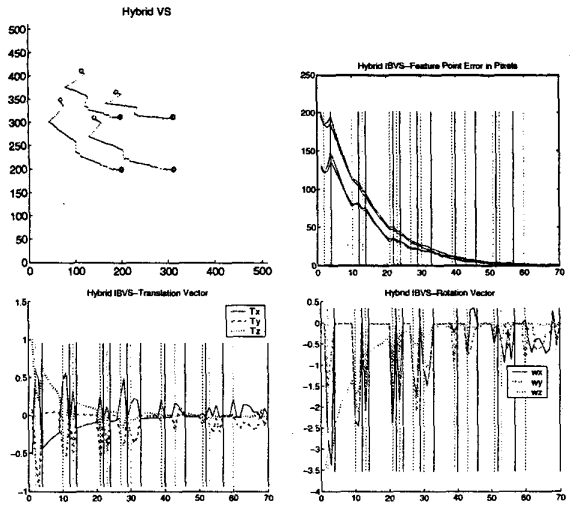


Fig. 10. General Motion with Noise, Random Switching

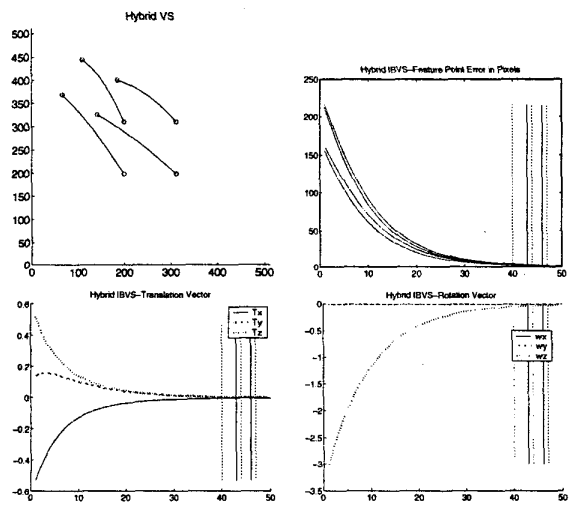


Fig. 11. Pure Affine Motion, Biased Random Switching

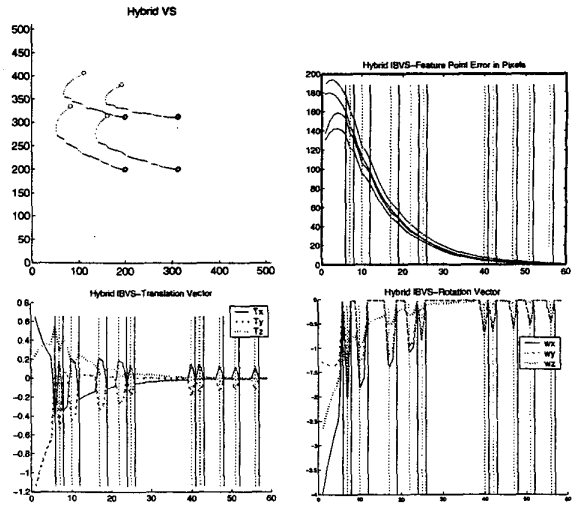


Fig. 12. General Motion, Biased Random Switching

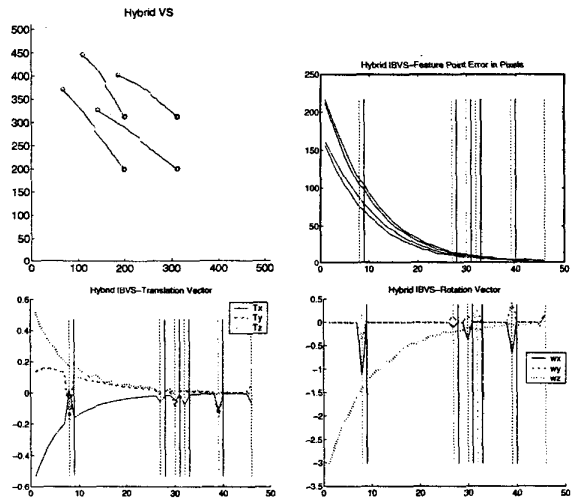


Fig. 13. Pure Affine Motion with Noise, Biased Random Switching

begin to surface as the controller oscillates between methods, with a slight preference for the affine method. The error is zeroed in just under 60 iterations, and the error plots look quite smooth. In contrast, the velocities are quite rough; each switch is accompanied by a harsh switch in velocities. However, the overall trajectory is fairy smooth. Figure 13 shows the effects of perturbing the feature point locations with noise. The system uses the affine method for almost the entire motion. Indeed it switches into the homography-based method fewer times than the deterministic switching system during the same test. The error is zeroed in about 45 iterations. Finally, Figure 14 shows system performance during a general motion with feature point locations perturbed by noise. The system switches often, but the error is quickly zeroed in under 60 iterations. The velocities are extremely ragged, particularly following a switch between methods.

VI. Future Research

This paper represents an initial step upon a future path in visual servoing. We have demonstrated the potential of a hybrid, switched control system, but much work remains.

We have not yet addressed in any mathematically rigorous way the stability of our switched controller. Stability imposes requirements on both the switching rules and on the low-level controllers. It is known that a bad choice of switching rule can lead to an unstable system even if all of the controllers are stable. Likewise, a good choice of switching rule can sometimes stabilize a set of unstable controllers. In the future, we will evaluate system stability from both theoretical and an empirical perspectives. For the former, we will exploit techniques from the hybrid dynamical systems literature together with existing results concerning the stability of certain existing visual servo controllers.

We have recently begun experimental verification of our sys-

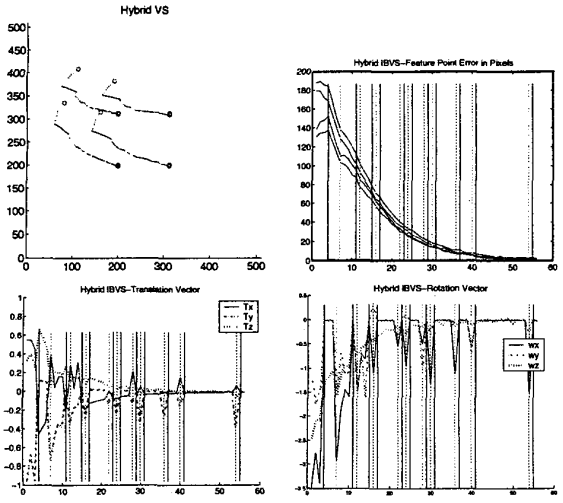


Fig. 14. General Motion with Noise, Biased Random Switching

tem. Using a PUMA 560 robot and Sony VFW-L500 digital camera. Results are still very preliminary, but promising. Figure 15 shows the pixel error and 6-tuple velocity vector for a large general motion. Both a random switching and determistic switching method are presented. Note that while the random switching takes longer to converge, the max error is much less. Indeed the deterministic method almost loses the feature points. Future, in depth, exploration will shine light on the capabilities of this system and switching systems in general.

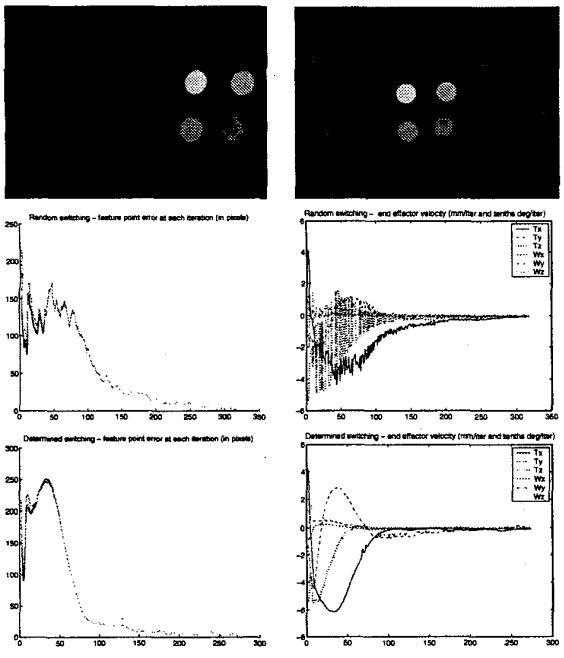


Fig. 15. General Motion with Noise, Biased Random Switching

VII. Conclusion

We have presented a novel hybrid visual servo system comprised of two individual systems and a decision maker to switch between them. Both systems forego the use of an image Jacobian and solve the transform relating initial and goal images to provide a camera trajectory. To test the feasibility of this hybrid system, we developed three potential switching rules based on input data, random choice, and a probabilistic blending of the two. Performance of each switching system was tested under a variety of control tasks and in the input of noise. For each test the system successfully zeroed the feature point error. These results are persuading, but only represent the beginning of exploration. Future goals include confirmation of stability, and generation of additional hybrid systems.

Additionally, we eschewed the use of the image Jacobian in our control systems, creating two systems that generate Cartesian camera trajectories based on data from two images. While the image Jacobian has proven invaluable, expanding and diversifying the pool of existing visual servo methods should not be neglected. Our goal is to proffer these additional avenues of study that hold a great deal of potential visual servo control.

References

[1] 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*, vol. RA-3, pp. 404–417, Oct. 1987.

[2] J. Feddema and O. Mitchell, "Vision-guided servoing with feature-based trajectory generation," *IEEE Trans. on Robotics and Automation*, vol. 5, pp. 691–700, Oct. 1989.

[3] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. on Robotics and Automation*, vol. 8, pp. 313–326, June 1992.

[4] 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. on Robotics and Automation*, vol. 9, no. 1, pp. 14–35, 1993.

[5] F. Chaumette, "Potential problems of stability and convergence in image-based and position-based visual servoing," in *The confluence of vision and control* (D. Kriegman, G. Hager, and S. Morse, eds.), vol. 237 of *Lecture Notes in Cont. and Info. Sci.*, pp. 66–78, Springer-Verlag, 1998.

[6] E. Malis, F. Chaumette, and S. Boudet, "2-1/2d visual servoing," *IEEE Trans. on Robotics and Automation*, vol. 15, pp. 238–250, Apr. 1999.

[7] G. Morel, T. Liebezeit, J. Szewczyk, S. Boudet, and J. Pot, "Explicit incorporation of 2d constraints in vision based control of robot manipulators," in *Experimental Robotics VI* (P. Corke and J. Trevelyan, eds.), vol. 250 of *Lecture Notes in Cont. and Info. Sci.*, pp. 99–108, Springer-Verlag, 2000.

[8] K. Deguchi, "Optimal motion control for image-based visual servoing by decoupling translation and rotation," in *Proc. Int. Conf. Intelligent Robots and Systems*, pp. 705–711, Oct. 1998.

[9] P. I. Corke and S. A. Hutchinson, "A new partitioned approach to image-based visual servo control," in *Proc. on 31st Int'l Symposium on Robotics and Automation*, 1999.

[10] N. R. Gans, P. I. Corke, and S. A. Hutchinson, "Performance tests of partitioned approaches to visual servo control," in *Proc. IEEE Int'l Conf. on Robotics and Automation*, 2002.

[11] M. Branicky, V. Borkar, and S. Mitter, "A unified framework for hybrid control," in *Proc. of the 33rd IEEE Conf. on Decision and Control*, 1994.

[12] R. W. Brockett, *Hybrid models for motion control systems*. 1993. H. L. Trentelman and J. C. Willems, Eds.

[13] D. Liberzon and A. Morse, "Basic problems in stability and design of switched systems," *IEEE Control Systems Magazine 19*, 1999.

[14] M. Branicky, "Multiple lyapunov functions and other analysis tools for switched and hybrid systems," in *IEEE Trans. Automat. Contr.*, 1998.

[15] O. Faugeras and F. Lustman, "Motion and structure from motion in a piecewise planar environment," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 2, no. 3, pp. 485–508, 1988.

[16] Z. Zhang and A. Hanson, "3d reconstruction based on homography mapping," in *ARPA Image Understanding workshop, Palm Springs, CA*. 1996.

[17] R. Boel and J. van Schuppen, "Distributed routing for load balancing," in *Proceedings of the IEEE, vol.77, Iss.1, 1989*, pp. 210–221, jan 1989.