Proceedings of the 8th IEEE Mediterranean Conference
on Control and Automation (MED 2000)
Rio, Patras, GREECE 17-19 July, 2000

A New Partitioned Approach to Image-Based Visual Servo Control

Peter I. Corke
CSIRO Manufacturing Science & Technology
Pinjarra Hills
AUSTRALIA 4069.
pic@cat.csiro.au

Seth A. Hutchinson
Beckman Institute for Advanced Technology
University of Illinois at Urbana-Champaign
Urbana, Illinois, USA 61801
seth@uiuc.edu

Abstract

In image-based visual servo control, since control is effected with respect to the image, there is no direct control over the Cartesian velocities of the robot end effector. As a result, trajectories that the robot executes, while producing image trajectories that are pleasing, can be quite contorted in the Cartesian space.

In this paper we introduce a new partitioned approach to visual servo control that overcomes this problem. In particular, we decouple the z-axis rotational and translational components of the control from the remaining degrees of freedom. Then, to guarantee that all features remain in the image throughout the entire trajectory, we incorporate a potential function that repels feature points from the boundary of the image plane. We illustrate our new control scheme with a variety of results.

1 Introduction

Broadly speaking, there are two basic approaches to visual servo control: Image-Based Visual Servo (IBVS), and, Position-Based Visual Servo (PBVS) [6]. In IBVS, an error signal is measured in the image, and is mapped directly to actuator commands. In PBVS systems, features are extracted from an image, and subsequently used to compute a (partial) 3D reconstruction of the environment or of the motion of a target object in the environment. An error is then computed in the task space, and it is this error that is used by the control system. Thus, the actual control problem confronted by a PBVS system is the classical robotics problem of tracking a Cartesian trajectory.

IBVS approaches have seen increasing popularity, largely due to the shortcomings of PBVS systems. With PBVS, any errors in calibration of the vision system will lead to errors in the 3D reconstruction, and subsequently to errors during task execution. In addition, since the control law for PBVS is defined in terms of the 3D workspace, there is no mechanism by which the image is directly regulated. Thus it is possible that objects of interest (including features used by the visual servo system) can exit the camera's field of view.

There are, however, also problems associated with IBVS systems. For an IBVS system the control law involves the mapping between image space velocities and velocities in the robot's workspace. This mapping is encoded in the image Jacobian, and, as one would expect, singularities in this Jacobian (which occur as a function of the relative position and motion of the camera and the object under observation) lead to control problems. This is, perhaps, the most persistent problem arising in IBVS systems. Second, since control is effected with respect to the image, there is no direct control over the Cartesian velocities of the robot end effector. Thus, trajectories that the robot executes, while producing visually appealing images, can appear contorted in the Cartesian space.

These performance problems with IBVS systems have led to the recent introduction of several hybrid methods [3,7,8]. Hybrid methods use IBVS to control certain degrees of freedom while using other techniques to control the remaining degrees of freedom. In Section 4 we describe a number of these hybrid approaches, and how they address specific performance issues. Then, in Section 5 we present a new partitioned visual servo control scheme that overcomes a number of the performance problems faced by previous systems. Finally, in Section 6, we describe how artificial potential fields defined in the image space can be used to enforce feature visibility.

2 Traditional IBVS

Let $r = (x, y, z)^T$ represent coordinates of the end-effector, and $\dot{r} = (T_x, T_y, T_z, \omega_x, \omega_y, \omega_z)^T$ represent the corresponding end-effector velocity. Let $f = (u, v)^T$ be the image-plane coordinates of a point in the image and $\dot{f} = (\dot{u}, \dot{v})^T$ the corresponding velocities. The image Jacobian relationship is given by

$$\dot{f} = J(r)\dot{r}, \tag{1}$$

with

$$J = \begin{bmatrix} \frac{\lambda}{z} & 0 & \frac{-u}{z} & \frac{-uv}{\lambda} & \frac{\lambda^2+u^2}{\lambda} & -v \\ 0 & \frac{\lambda}{z} & \frac{-v}{z} & \frac{-\lambda^2-v^2}{\lambda} & \frac{uv}{\lambda} & u \end{bmatrix} \quad (2)$$

in which λ is the focal length for the camera. Derivations of this can be found in a number of references including [6]. Eq. (1) can be decomposed, and written as

$$\dot{f} = J_v(u, v, z)\mathbf{v} + J_\omega(u, v)\omega, \quad (3)$$

in which $J_v(u, v, z)$ contains the first three columns of the image Jacobian, and is a function of both the image coordinates of the point and its depth, while $J_\omega(u, v)$ contains the last three columns of the image Jacobian, and is a function of only the image coordinates of the point (i.e., it does not depend on depth). This decomposition is at the heart of the hybrid methods that we discuss below.

The simplest approach to IBVS is to merely use (1) to construct the control law

$$\mathbf{u} = \Gamma J^{-1}(r)\dot{f} \quad (4)$$

in which $\dot{f}$ is the desired feature motion on the image plane, Γ is a gain matrix, and $\mathbf{u} = \dot{r}$ is the control input, an end-effector velocity (this can be converted to joint velocities via the manipulator Jacobian). Of course this approach assumes that the image Jacobian is square and nonsingular, and when this is not the case, a generalized inverse, J^+, is used. Since (4) essentially represents a gradient descent on the feature error, when this control law is used, feature points move in straight lines to their goal positions. This can be seen in Figure 1(a).

3 Performance Issues

A commonly mentioned criticism of IBVS is that the Cartesian paths often involve large camera motions, which are undesirable. Often the camera moves away from the target in a normal direction and then returns, a phenomenon we refer to as *camera retreat*, which is illustrated in Figure 1. In Figure 1(a), the feature points are driven on straight line trajectories to their goal positions, producing a large, and seemingly unnecessary, motion in the z-direction, seen in Figure 1(c).

In [1], Chaumette introduced an extreme version of this problem, which we refer to as the Chaumette Conundrum, illustrated in Figure 2. Here, the desired camera pose corresponds to a pure rotation about the optic axis by π rad. Since control laws of the form given in (4) drive the feature points in straight lines, the feature points are driven toward the origin, which corresponds to a singularity in the image Jacobian. As noted in [1],

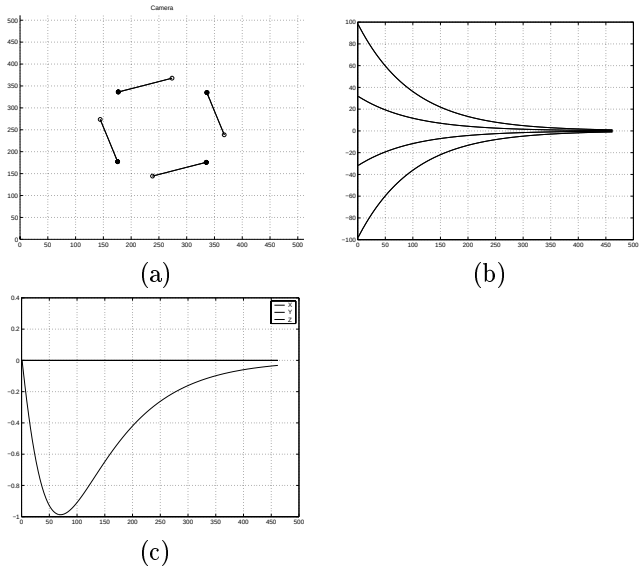


Figure 1: IBVS for pure target rotation (0.3 rad). (a) Image-plane feature motion (initial location is $\circ$, desired location is $\bullet$), (b) Feature error trajectory, (c) Cartesian translation trajectory.

this problem cannot be detected by simply examining the image Jacobian, since the image Jacobian is well conditioned (at least initially).

At first it might seem that some rotational motion of the camera about its optic axis should be induced for the Chaumette Conundrum; however, this is not the case. The ω_z component of (4) is given by

$$\omega_z = (J^+)_6\dot{f} \quad (5)$$

in which $(J^+)_6$ denotes the bottom row of the generalized inverse. In this particular case, even though $\dot{f} \neq 0$, the inner product is zero, i.e., the various contributions to rotational velocity cancel one another.

This camera retreat phenomenon can be explained in geometric terms, leading to a simple model that predicts the magnitude of the camera retreat motion. For the example of Figure 1, a pure rotational motion of the camera would cause the points to follow an arc from point A to point B, as shown in Figure 3. For the points to follow a straight line, as specified by (4), the scale must be changed so as to move the point from B to C. The required change in scale is given simply by the ratio of the distances OC and OB. The scale reduction attains its maximum value at $\theta = \alpha/2$ for which

$$\left(\frac{OC}{OB}\right)_{max} = \cos\frac{\alpha}{2}. \quad (6)$$

In the IBVS the reduction in scale is achieved by moving the camera away from the target. The reduction

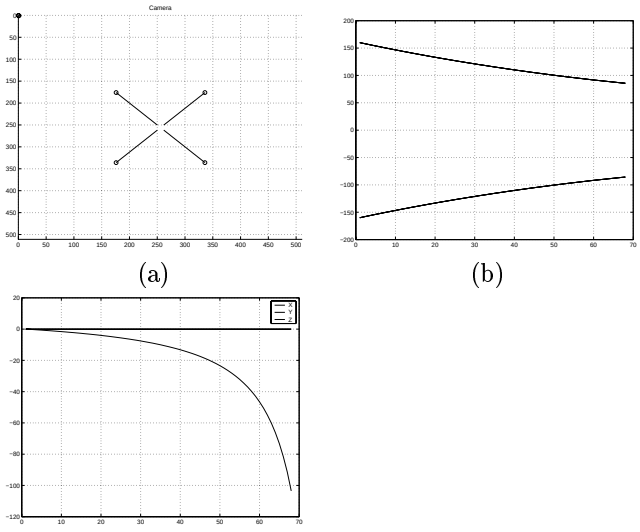


Figure 2: Performance of classical IBVS with the Chaumette example. (a) Image-plane feature motion (initial location is ∘, desired location is •), (b) Feature error trajectory, (c) Cartesian translation trajectory.

in the apparent length of the line segment is inversely proportional to the distance that the camera retreats, and therefore,

$$\frac{OC}{OB} = \frac{d_{targ}}{d} \tag{7}$$

in which d is the current distance to the target, and d_{targ} is the desired target distance, and assuming the camera is moving normal to the target. The maximum reduction is thus given by

$$\frac{d_{targ}}{d_{max}} = \cos\frac{\alpha}{2}. \tag{8}$$

For the Chaumette Conundrum, in which $\alpha = \pi$, the model accurately predicts infinite camera retreat. The maximum camera retreat ratio observed in visual servo simulations and the simple model of (8) are compared in Figure 4 and show close agreement.

At first it might seem that the introduction of line segment features would solve the problem, since the orientation of such a segment is unambiguous. Chaumette notes that such an approach is not guaranteed to solve the performance problems [1], and our own simulation results support this conclusion. Specifically, in simulations we added one extra row to the image Jacobian corresponding to a line segment angle feature [2]. Its effect was not significant. For the Chaumette Conundrum the addition of this feature does command *some* camera rotation, but this commanded rotational motion is nearly 3 orders of magnitude less than the Z-axis translation, even when scaling of feature magnitudes is taken into account.

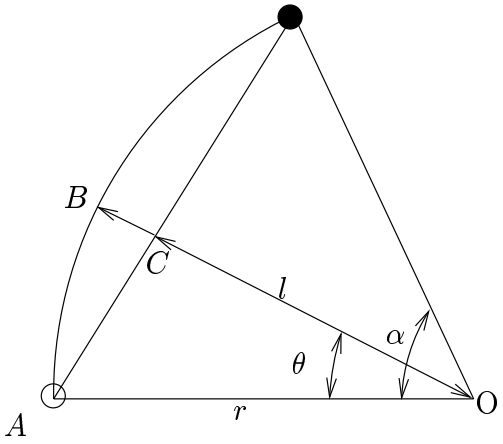


Figure 3: Camera retreat model.

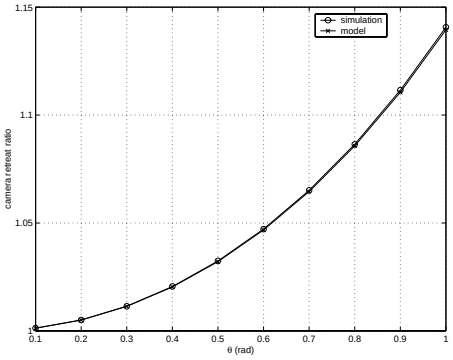


Figure 4: Comparison of camera retreat ratios.

One approach to the camera retreat problem is to decouple the z-axis translational and rotational motions from the control law of (4). Separate controllers would then be designed to enforce appropriate rotational and retreat motions. This leads to hybrid approaches that combine aspects of IBVS and PBVS systems. In Section 4 we describe several such approaches that have been recently introduced. Then, in Section 5 we introduce our new partitioned method.

4 Some Recent Hybrid Approaches

The approaches described in [3,7,8] propose hybrid control architectures that exploit the decomposition of (3). In particular, the camera configurations that correspond to the initial and desired images are related by a homography matrix, which can be computed from a set of corresponding points in the initial and desired images [5]. As has been shown in [4], for the special case of four coplanar points, this homography can be decomposed into a rotational component and a translational com-

ponent. The translational component can be recovered only up to scale, and therefore, depth must be estimated if the translational component is to be used in a visual servo scheme (as is the case for [3]).

With 2.5-D visual servo [7], IBVS is used to control translational degrees of freedom, while the homography matrix is used to estimate the desired rotational motion. The significant innovation in the 2.5-D visual servo method is their novel method for controlling the camera's rotational DOF. In [7], orientation is expressed as a rotation, θ, about an axis, $\mathbf{u}$. The resulting control is given by

$$\omega = \mathbf{u}\theta, \quad \mathbf{v} = -\lambda J_v^{-1}\dot{f} + \lambda J_v^{-1} J_\omega \mathbf{u}\theta. \tag{9}$$

Thus, the rotational component of the control is computed directly from the computed desired rotation in 3D, and the translational component is computed by subtracting from the traditional IBVS control a term that accounts for the translational motion in the image induced by the rotation. Results in [1] and [7] show that this new method handles the problem of Figure 1 and eliminates camera retreat.

The only drawback to this approach seems to be that the commanded rotational motions may cause feature points to leave the image-plane. With pure IBVS, the paths of the feature points are straight lines in the image plane, and this problem does not arise if the points are visible at the start and end of the motion (if the viewable portion of the image plane is convex).

The problem of feature points leaving the image plane during 2.5-D visual servo motion motivated Morel et al. to propose a modified approach [8]. They use the same control as given by (9), but use a different feature vector (and, accordingly, an appropriate image Jacobian).

Deguchi takes the opposite approach from the 2.5-D scheme of Malis et al. [3]. In particular, he uses the decomposition of the homography matrix to compute the translation velocity, leading to the control

$$\mathbf{v} = \hat{d}\left(\frac{t}{d^*}\right), \quad \omega = -\lambda J_\omega^{-1}\dot{f} + \lambda J_\omega^{-1} J_v \mathbf{v}. \tag{10}$$

Here, $\hat{d}$ is the estimated depth of the point in 3D and the ratio t/d^* is the scaled translation that is directly yielded by the decomposition of the homography matrix. Thus, the translational component of the control is computed directly from the estimated desired translation in 3D, and the rotational component is computed by subtracting from the traditional IBVS control a term that accounts for the motion in the image that is induced by the translation.

Deguchi also presents a second method, in which the essential matrix [5] is used (instead of the homography matrix) to compute the desired translational component. This method yields essentially the same control as the first method, but the constraint that the four feature points be coplanar is removed.

5 A New Partitioned IBVS Scheme

Our approach is based on the observation that while IBVS works well for small motions, problems arise with large motions and particularly involving rotation about the z axis. Our scheme singles out Z-axis motion for special treatment, motivated by the fact that the performance issues we confront are directly related to Z-axis translation and rotation. We partition (1) so that

$$\dot{f} = J_{xy}\dot{r}_{xy} + J_z\dot{r}_z \tag{11}$$

where $\dot{r}_{xy} = [Tx\ Ty\ \omega_x\ \omega_y]$, $\dot{r}_z = [T_z\ \omega_z]$, and J_{xy} and J_z are respectively columns $\{1,\ 2,\ 4,\ 5\}$ and $\{3,\ 6\}$ of J. We can write (11) as

$$\dot{r}_{xy} = J_{xy}^{+}\left\{\dot{f} - J_z\dot{r}_z\right\} \tag{12}$$

where $\dot{f}$ is the feature point coordinate error as in the traditional IBVS scheme.

The Z-axis velocity, $\dot{r}_z$, is based directly on two new image features that are simple and inexpensive to compute. The first image feature, $0 \le \theta_{ij} < 2\pi$, is the angle between the u-axis of the image plane and the directed line segment joining feature points i and j. For numerical conditioning it is advantageous to select the longest line segment that can be constructed from the feature points. The rotational rate is simply

$$\omega_z = \gamma_{\omega_z}(\theta_{ij}^* - \theta_{ij})$$

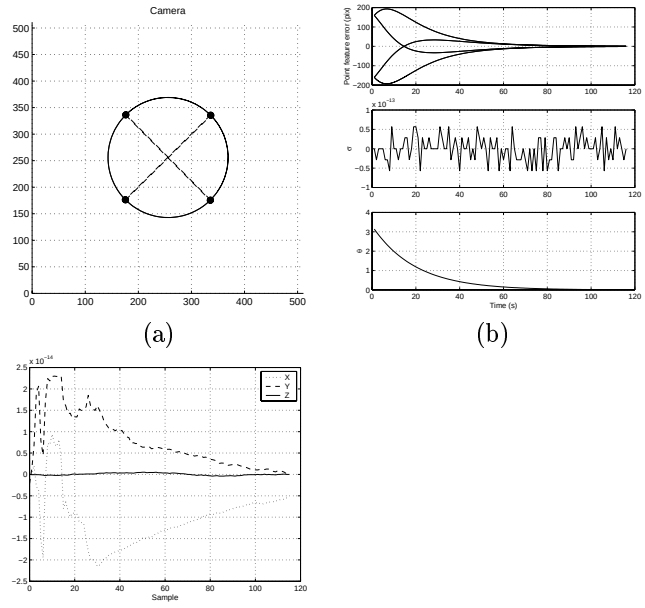
in which γ_{ω_z} is a scalar gain coefficient.

The second new image feature that we use, σ, is the square root of the area of the regular polygon whose vertices are the image feature points. The camera z-axis translation rate is given by

$$T_z = \gamma_{T_z}(\sigma^* - \sigma). \tag{13}$$

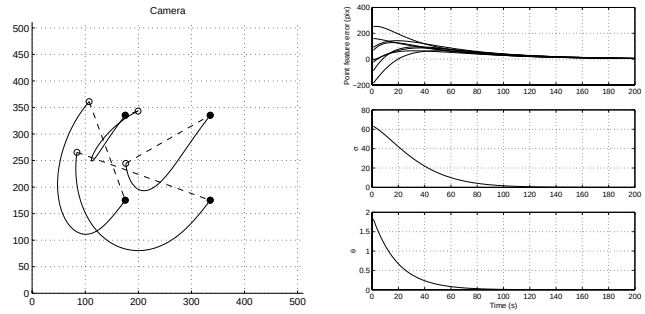
Figure 5 shows the performance of the proposed partitioned controller for the Chaumette Conundrum. The important features are that the camera does not retreat since σ is constant. The rotation θ monotonically decreases and the feature points move in a circle. The feature coordinate error is initially increasing, in contrast to classical IBVS.

An example that involves more complex translational and rotational motion is shown in Figure 6. The new features decrease monotonically, but the error in f does not decrease monotonically and the points follow complex curves on the image plane. Figure 7 compares the



(a) (b) (c)

Figure 5: Proposed partitioned IBVS for pure target rotation (π rad). (a) Image-plane feature motion (initial location is $\circ$, desired location is $\bullet$), (b) Feature error trajectory, (c) Cartesian translation trajectory.



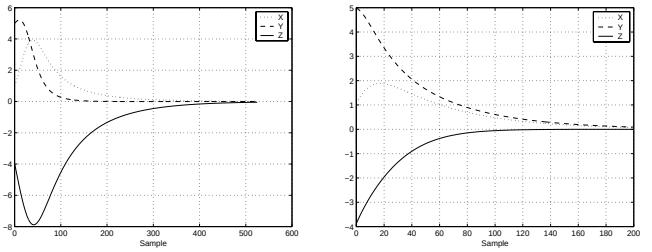
(a) (b)

Figure 6: Proposed partitioned IBVS for general target motion. (a) Image-plane feature motion (dashed line shows straight line motion for classical IBVS), (b) Feature error trajectory.

Cartesian camera motion for the two IBVS methods. The proposed partitioned method has eliminated the camera retreat and also exhibits better behavior for the X- and Y-axis motion.

6 Enforcing Feature Visibility

In order to keep all feature points inside the viewable portion of the image plane at all times, we borrow collision avoidance techniques from the robot motion planning community. In particular, we establish a repulsive potential at the boundary of the viewable portion of the image, and incorporate the gradient of this potential into the control law. We use the simple potential given by



IBVS Proposed partitioned IBVS

Figure 7: Comparison of Cartesian camera motion for classic and new partitioned IBVS for general target motion.

$$U_{rep}(u,v) = \begin{cases} \frac{1}{2}\eta\left(\frac{1}{\rho(u,v)} - \frac{1}{\rho_0}\right) & : \quad \rho(u,v) \leq \rho_0 \\ 0 & : \quad \rho(u,v) > \rho_0 \end{cases} \tag{14}$$

in which $\rho(u,v)$ is the shortest distance to the edge of the image plane from the image point with coordinates (u,v). The value ρ_0 specifies the zone of the image in which U_{rep} affects the control; if the feature point is not with distance ρ_0 of the boundary, then the corresponding motion is not affected by U_{rep}. The value of η is a scalar gain coefficient.

For an $N_r \times N_c$ image, the value of ρ is easily computed as

$$\rho(u,v) = \min\{u, v, N_r - u, N_c - v\}. \tag{15}$$

If $\mathbf{n}$ is the unit vector directed from the nearest boundary to image feature point with coordinates (u,v), then $\nabla U_{rep} = F\mathbf{n}$, with F given by

$$F(u,v) = \begin{cases} \eta\left(\frac{1}{\rho(u,v)} - \frac{1}{\rho_0}\right)\frac{1}{\rho^2(u,v)} & : \quad \rho(u,v) \leq \rho_0 \\ 0 & : \quad \rho(u,v) > \rho_0 \end{cases}. \tag{16}$$

Since a pure translation in the negative z-direction will cause feature points to move toward the center of the image, the value of F is mapped directly to the T_z component of the velocity command by combining it with the control given in (13). Because of chatter effects (where the feature points oscillate in and out of the potential field), we smooth and clip the resulting T_z, yielding the discrete-time controller

$$T_z'(k) = \mu T_z'(k-1) + (1-\mu)(\sigma^\star - \sigma - F) \tag{17}$$

$$T_z = \min\{\max\{T_z'(k), T_{z_{min}}\}, T_{z_{max}}\}. \tag{18}$$

In simulation we found it advantageous to use asymmetric velocity clipping, $|T_{z_{max}}| < |T_{z_{min}}|$, i.e., the camera can retreat faster than it can approach the target. This reduces the magnitude of the "bounces" off the image plane boundaries when points first enter the potential field. In practice it may not be necessary to explicitly implement smoothing and clipping, since real robots have finite bandwidth and velocity capability.

The use of a potential field raises the issue of local minima in the field, but in our case, these issues do not arise, since the potential field is used merely to force a camera retreat, and since it will be possible for the system to achieve the goal when this retreat is effected (in this case we merely approach the performance of the classical IBVS system). This requires that no goal feature point locations lie within the influence of the potential field, which can be enforced by the choice of ρ_0.

Results of the new partitioned IBVS with collision avoidance are shown in Figure 8. The target is larger than before, so as the camera rotates the feature points move into the potential field. The parameters used were $\eta = 5 \times 10^6$ and $\mu = 0.8$. It can be seen that as the points are rotated, they move into the potential field and then follow a path parallel to the edge, where the repulsion and scale demand are in equilibrium.

For high rotational rates, the chatter phenomenon will occur, and at very high rates the points may pass through the potential field and become trapped *outside* the image plane. Rotational rate should properly be controlled by another loop, and this problem has strong similarities to that of controlling step size in numerical optimization procedures.

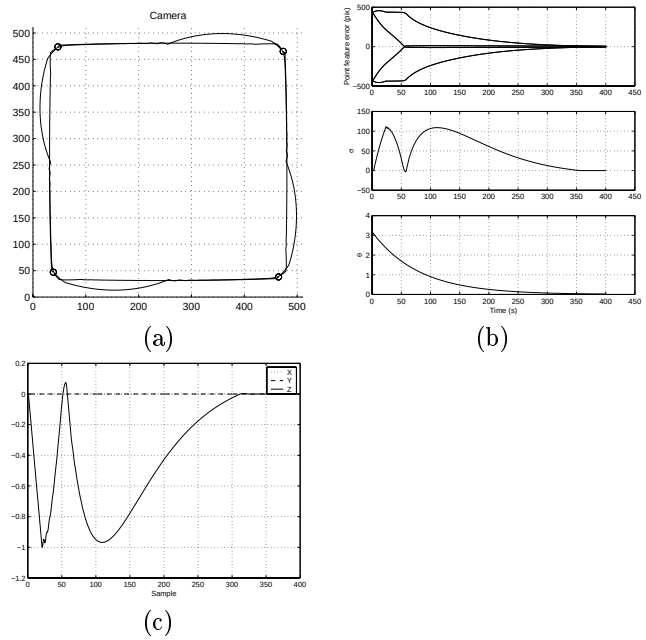


Figure 8: Proposed partitioned IBVS with collision avoidance for pure target rotation (π rad). (a) Image-plane feature motion (initial location is $\circ$, desired location is $\bullet$), (b) Feature error trajectory, (c) Cartesian translation trajectory.

7 Conclusion

We have investigated some problems with classical IBVS approaches and proposed a new partitioned visual servoing scheme that inexpensively overcomes these limitations. Other hybrid schemes have been proposed and are based on decoupling camera translational and rotational degrees of freedom. We have proposed a different decoupling and servo Z-axis rotation and translation using decoupled controllers based on two easily computed image features. All hybrid schemes admit the possibility of points leaving the image plane, as does the approach that we described in 5. We treat this as a collision avoidance problem and use potential field techniques to repel the feature points from the image plane boundary.

References

[1] F. Chaumette. Potential problems of stability and convergence in image-based and position-based visual servoing. In D. Kriegman, G. Hager, and S. Morse, editors, *The confluence of vision and control*, volume 237 of *Lecture Notes in Control and Information Sciences*, pages 66–78. Springer-Verlag, 1998.

[2] Francois Chaumette. *La relation vision-commande: théorie et application à des tâches robotiques.* PhD thesis, L'Univesité de Rennes I, 1990.

[3] K. Deguchi. Optimal motion control for image-based visual servoing by decoupling translation and rotation. In *Proc. Int. Conf. Intelligent Robots and Systems*, pages 705–711, October 1998.

[4] O. D. Faugeras and F. Lustman. Motion and Structure from Motion in a Piecewise Planar Environment. *Int'l Journal of Pattern Recognition and Artificial Intelligence*, 2(3):485–508, 1988.

[5] O.D. Faugeras. *Three-Dimensional Computer Vision.* MIT Press, Cambridge, MA, 1993.

[6] S. Hutchinson, G. Hager, and P. Corke. A tutorial on visual servo control. *IEEE Trans. Robot. Autom.*, 12(5):651–670, October 1996.

[7] E. Malis, F. Chaumette, and S. Boudet. 2-1/2-d visual servoing. *IEEE Trans. Robot. Autom.*, 15(2):238–250, April 1999.

[8] G. Morel, T. Liebezeit, J. Szewczyk, S. Boudet, and J. Pot. Explicit incorporation of 2d constraints in vision based control of robot manipulators. In Peter Corke and James Trevelyan, editors, *Experimental Robotics VI*, volume 250 of *Lecture Notes in Control and Information Sciences*, pages 99–108. Springer-Verlag, 2000. ISBN: 1 85233 210 7.