

New shortest-path approaches to visual servoing

Ville Kyrki
Laboratory of Information Processing
Lappeenranta University of Technology
Lappeenranta, Finland
kyrki@lut.fi

Danica Kragic and Henrik I. Christensen
Center for Autonomous Systems
Royal Institute of Technology
Stockholm, Sweden
{danik,hic}@nada.kth.se

Abstract—In recent years, a number of visual servo control algorithms have been proposed. Most approaches try to solve the inherent problems of image-based and position-based servoing by partitioning the control between image and Cartesian spaces. However, partitioning of the control often causes the Cartesian path to become more complex, which might result in operation close to the joint limits. A solution to avoid the joint limits is to use a shortest-path approach, which avoids the limits in most cases.

In this paper, two new shortest-path approaches to visual servoing are presented. First, a position-based approach is proposed that guarantees both shortest Cartesian trajectory and object visibility. Then, a variant is presented, which avoids the use of a 3D model of the target object by using homography based partial pose estimation.

I. INTRODUCTION

Visual servo control of robot motion has been introduced to increase the accuracy and application domain of robots. Closing the loop using vision allows the systems to control the pose of a robot relative to a target also in the case of calibration and measurement errors.

The development of different servoing strategies has received a considerable amount of attention. In addition to the convergence of the control law, the visibility of the servoing target needs to be guaranteed to build a successful system. To guarantee the visibility of the target, many of the proposed control strategies control the motion in a partitioned task space, where the task function of the control is defined in terms of both image and 3D features.

A characteristic that is often neglected is the Cartesian trajectory of a servo control law. This is partly due to the partitioned control, which makes the analysis of the trajectory difficult. However, the knowledge of the Cartesian trajectory is essential to avoid the joint limits.

We are interested in shortest path servoing, because it is predictable and its consequent straight line trajectory avoids trying to move outside the robot workspace in most cases. We present a visual servoing strategy which accomplishes two goals: it uses straight line trajectory minimizing the path length and also keeps the target object in the field of view at all times.

The rest of the paper is organized as follows: The research is first motivated by surveying related work in Section II. In Section III, position-based visual servoing is outlined. Section IV presents our servoing strategy for position-based control. In Section V the same strategy is applied to servoing based on a planar homography

estimated from image features. The strategy is evaluated and compared to previous shortest path approaches in Section VI. Finally, the results are summarized and discussed in Section VII.

II. RELATED WORK

Vision-based robot control is traditionally classified into two types of systems [1]: position-based and image-based systems. In position-based visual servoing (PBVS), full 3D object pose is estimated using a model of the object, and the control error is then defined in terms of the current and desired poses [2]. PBVS makes it possible to attain straight line motion resulting in a minimum trajectory length in 3D. The computational complexity of pose estimation has been earlier considered a disadvantage of this method. Another well-known problem, pointed out by Chaumette [3], is that PBVS poses no constraints on the visibility of the object. That is, the object which is used for pose estimation may move outside the field of view of the camera.

In image-based visual servoing (IBVS), the control error is expressed in the 2D image plane. This error is then transformed into Cartesian space using the inverse of image Jacobian. While this guarantees the visibility for all feature points, it results in convergence problems due to local minima of the error function and possible singularities of the image Jacobian [3]. In addition, the Cartesian trajectory is very unpredictable, which may result in reaching the joint limits. This approach was supposed to be tolerant against errors in depth estimates required to estimate the image Jacobian. However, a recent study shows that the convergence region is not very large [4].

Recently, a new group of methods has been proposed, called partitioned visual servoing. These address some of the problems mentioned above by combining traditional image Jacobian based control with other techniques. Chaumette et al.[5], [6] have proposed a method called 2.5D visual servoing, which decouples translation and rotation control. It addresses the visibility problem by controlling the movement of one feature point in the image plane, and does not require a full 3D model as a planar homography is used for partial pose estimation. The Cartesian trajectory is likely to be reasonable as the range to the object is controlled, but nevertheless not very easy to predict.

Deguchi [7] has proposed a decoupled control scheme which uses planar homography or epipolar geometry for

partial pose estimation. The translation is controlled along the shortest path in Cartesian coordinates, while the rotation is controlled using the image Jacobian. A problem of this method is that the Jacobian may suffer from singularities as in the case of 180 degree rotation around the optical axis [8].

Corke and Hutchinson [9] have noticed that an IBVS problem of camera retreat can be solved by decoupling the z-axis translation and rotation from the image Jacobian and propose to control them using simple image features. They also address the visibility problem by using a repulsive potential function near image edges. However, there is no knowledge about the path length and the image error function might have multiple local minima.

In addition to the partitioned approaches mentioned, Mezouar and Chaumette [10] have combined pure IBVS with potential field based path planning. Path-planning makes it possible to use IBVS only over short transitions which solves most problems. This makes it possible to pose constraints to guarantee the visibility of features and to avoid robot joint limits. However, it is not clear if the potential field may have local minima, where the attractive and repulsive forces cancel each other, and the trajectory is hard to predict.

Gans and Hutchinson [11] have proposed another possible approach, namely switching between IBVS and PBVS. Thus, whenever the visibility problem of PBVS is imminent, the control is switched to IBVS. If camera retreat occurs, the control is again switched to PBVS. The system has been shown to be asymptotically stable, but again the trajectories are difficult to predict.

III. POSITION BASED VISUAL SERVOING

In position-based visual servoing (PBVS), the task function is defined in terms of the displacement-vector from the current to the desired position, which can be expressed using the transformation ${}^cT_{c^*}$. The input image is usually used to estimate the object to camera transformation cT_o which can be composed with the desired pose to object transformation ${}^oT_{c^*}$ to find the relation from the current to the desired pose. By decomposing the transformation matrices into translation and rotation, this can be expressed as

$$\begin{aligned} {}^cT_{c^*} &= {}^cT_o {}^oT_{c^*} = \begin{pmatrix} {}^cR_o & {}^c\mathbf{t}_o \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} {}^oR_{c^*} & {}^o\mathbf{t}_{c^*} \\ \mathbf{0} & 1 \end{pmatrix} \\ &= \begin{pmatrix} {}^cR_o {}^oR_{c^*} & {}^cR_o {}^o\mathbf{t}_{c^*} + {}^c\mathbf{t}_o \\ \mathbf{0} & 1 \end{pmatrix} = \begin{pmatrix} {}^cR_{c^*} & {}^c\mathbf{t}_{c^*} \\ \mathbf{0} & 1 \end{pmatrix} \end{aligned} \quad (1)$$

Then, the translation vector to the desired position in camera coordinates is ${}^c\mathbf{t}_{c^*}$. For the orientation the rotation matrix can be decomposed into axis of rotation \mathbf{u} and angle θ , which can be multiplied to get the desired rotational movement $\mathbf{u}\theta$.

Thus, the proportional control law can be defined as

$$\mathbf{v} = \lambda \begin{pmatrix} {}^c\mathbf{t}_{c^*} \\ \mathbf{u}\theta \end{pmatrix} \quad (2)$$

where λ is the gain factor.

This approach has minimum length trajectory in Cartesian coordinates. It may, however, lose the visibility of the target used for positioning.

IV. A NEW POSITION-BASED APPROACH

We present a new approach that has minimum length trajectory in Cartesian coordinates but does not suffer from the visibility problem.

The control function is now defined separately for translation and rotation. To ensure the shortest path, we use the position based control directly for translation. The three axes of rotation are controlled as follows: The rotation around the camera x - and y -axes is controlled using a virtual point located at the origin of the object frame. We control this point in such a way that it will always be visible which ensures the visibility of the target object used for positioning, provided that the distance to the target is adequate. Without loss of generality, we can assume unit focal length. Let ${}^c\mathbf{p}_o = ({}^cX_o, {}^cY_o, {}^cZ_o)^T$ be the coordinates of the object frame origin in the camera frame. That is, ${}^c\mathbf{p}_o$ corresponds to ${}^c\mathbf{t}_o$. Then, the coordinates of this point in the image plane (x, y) are

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{{}^cZ_o} \begin{pmatrix} {}^cX_o \\ {}^cY_o \end{pmatrix}. \quad (3)$$

Now, we can find the image velocity of the point with respect to the 3D velocity screw of the camera $\dot{\mathbf{q}}$ as

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} -\frac{1}{{}^cZ_o} & 0 & \frac{x}{{}^cZ_o} & xy & -(1+x^2) & y \\ 0 & -\frac{1}{{}^cZ_o} & \frac{y}{{}^cZ_o} & (1+y^2) & -xy & -x \end{pmatrix} \dot{\mathbf{q}}. \quad (4)$$

We will now control this point towards the origin, such that in the final position the origin of the object lies on the optical axis. Note that this does not constrain the positioning task, as the origin can be selected arbitrarily. This allows us to control two-degrees of rotational motion. The final rotational degree of freedom, namely the rotation around the optical axis, is controlled using the standard position based control law, that is, $u_z\theta$, the rotation around the optical axis is driven to zero. Now, we can define the task vector as $\mathbf{e} = ({}^cX_*, {}^cY_*, {}^cZ_*, x, y, u_z\theta)^T$, the first three terms denoting components of the desired position in the current camera coordinate frame ${}^c\mathbf{t}_{c^*}$. Using a simplified notation $(X, Y, Z)^T = ({}^cX_o, {}^cY_o, {}^cZ_o)^T$, the Jacobian of the task function can be written

$$\begin{pmatrix} {}^c\dot{X}_* & {}^c\dot{Y}_* & {}^c\dot{Z}_* & \dot{x} & \dot{y} & u_z\dot{\theta} \end{pmatrix}^T = \mathbf{J}\dot{\mathbf{q}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ -\frac{1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & -\frac{1}{Z} & \frac{y}{Z} & (1+y^2) & -xy & -x \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \dot{\mathbf{q}}, \quad (5)$$

or in the terms of the object position

$$\begin{pmatrix} c\dot{X}_* & c\dot{Y}_* & c\dot{Z}_* & \dot{x} & \dot{y} & u_z\theta \end{pmatrix}^T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ -\frac{1}{Z} & 0 & \frac{X}{Z^2} & \frac{XY}{Z^2} & -(1 + \frac{X^2}{Z^2}) & \frac{Y}{Z} \\ 0 & -\frac{1}{Z} & \frac{Y}{Z^2} & (1 + \frac{Y^2}{Z^2}) & -\frac{XY}{Z^2} & -\frac{X}{Z} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \dot{\mathbf{q}} \quad (6)$$

Proportional control law can now be defined using the inverse of the Jacobian as

$$\mathbf{v} = -\lambda\dot{\mathbf{q}} = -\lambda\mathbf{J}^{-1}\mathbf{e} = \lambda \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ & & & \mathbf{J}_x^{-1} & & \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{e} \quad (7)$$

where

$$\mathbf{J}_x^{-1} = \begin{pmatrix} -\frac{XY}{ZP} & -\frac{Y^2+Z^2}{ZP} \\ -\frac{X^2+Z^2}{ZP} & -\frac{XY}{ZP} \\ \frac{Y}{P} & -\frac{X}{P} \\ -\frac{XY}{P} & -\frac{Y^2+Z^2}{P} \\ \frac{X^2+Z^2}{P} & \frac{XY}{P} \\ \frac{X}{Z} & \frac{Y}{Z} \end{pmatrix}^T \quad (8)$$

where $P = X^2 + Y^2 + Z^2$. From (7) it can be seen that all translational degrees of freedom as well as the rotation around the camera optical axis are decoupled.

The determinant of the Jacobian is

$$|\mathbf{J}| = \frac{X^2 + Y^2 + Z^2}{Z^2} \quad (9)$$

which is zero only when all X , Y , and Z are zero, that is when the camera is exactly at the object origin. Another degenerate case occurs when the object lies on the camera image plane. However, this configuration is also physically not valid.

V. HYBRID APPROACH

We now present a hybrid approach utilizing similar ideas as presented above, but this approach is not dependent on information about full 3D pose. Rotation and direction of translation are instead estimated by decomposing a homography matrix [12], [13].

Image homography \mathbf{H}_z maps projective points which lie on a plane between two views such that $\mathbf{p}_* \propto \mathbf{H}_z\mathbf{p}$. It can be estimated for example using the method proposed by Malis et al. [13]. Assuming a camera calibration matrix \mathbf{C} is available, the homography \mathbf{H} can be calculated in the camera coordinates from

$$\mathbf{H} = \mathbf{C}^{-1}\mathbf{H}_z\mathbf{C} \quad (10)$$

The homography matrix can be written in terms of a rotation matrix and a translational component as

$$\mathbf{H} = \mathbf{R} + \frac{\mathbf{t}}{d_*}\mathbf{n}_*^T \quad (11)$$

where \mathbf{n}_* is the normal to the plane in the goal position and d_* its distance. The homography can be decomposed into \mathbf{R} , \mathbf{t}/d_* and \mathbf{n}_* [12]. The decomposition gives two solutions and the correct one can be selected by using additional information or time consistency. Thus, the rotation matrix \mathbf{R} can be fully recovered, while the translation vector can be recovered only up to a scale.

The idea of the hybrid approach is to control translation to the estimated direction, while using a single image point to control two axes of rotation (around x and y) and control the final axis of rotation (around camera optical axis) using the rotation matrix. The translation is compensated in the rotations around x and y . Thus, the image point will have a straight line trajectory and will always remain in the field of view.

Considering the Jacobian in (5), in addition to the coordinates of an image point, its depth is needed to compensate for the translational velocity. However, if the distance to the plane in the desired position is known, this can be computed using only quantities calculated from the homography as [13]

$$\frac{1}{Z} = \frac{\mathbf{n}^T\mathbf{p}}{d_*(1 + \mathbf{n}^T\frac{\mathbf{t}}{d_*})} \quad (12)$$

where \mathbf{p} is a point on the plane in current camera coordinates, and $\mathbf{n} = \mathbf{R}\mathbf{n}_*$ is the plane normal in current camera position.

To summarize, the translation is controlled by

$$\mathbf{v}_t = \lambda_t \hat{d}_* \frac{c\mathbf{t}_{c*}}{d_*} \quad (13)$$

where \hat{d}_* is an estimate of the distance to the plane in the goal position, and λ_t is a gain factor. It should be noted that errors in the estimate only appear as a gain factor. Rotation around z -axis is controlled by

$$\omega_z = \lambda_{\omega_z} u_z \theta \quad (14)$$

where λ_{ω_z} is again the gain factor. Then, rotation around x - and y -axes is controlled by

$$\begin{pmatrix} \omega_x & \omega_y \end{pmatrix}^T = \mathbf{J}_{\omega_{xy}}^{-1} (\mathbf{s} - \mathbf{J}_t \mathbf{v}_t - \mathbf{J}_{\omega_z} \omega_z) \quad (15)$$

where \mathbf{s} is the error of the control point in the image. \mathbf{J}_t , $\mathbf{J}_{\omega_{xy}}$, and \mathbf{J}_{ω_z} are the Jacobians for translation, rotation in x and y , and rotation in z , respectively.

VI. EXPERIMENTS

We present the simulation experiments where the proposed method is compared to previously known shortest-path servoing methods, namely standard PBVS and the method proposed by Deguchi [7]. To compare the servoing method, we adopt the set of standard tasks and measures proposed by Gans et al.[8]. They propose the following control tasks: 1) Optical axis rotation, 2) optical axis translation, 3) camera y -axis rotation, and 4) feature plane rotation. In addition, we have a fifth control task which we call "general motion".

We adopt also the metrics proposed in [8]. The performance is evaluated based on 1) time to convergence,

2) maximum feature excursion, 3) maximum camera excursion, and 4) maximum camera rotation. To measure the time to convergence, a method was considered to be successful when the average feature point error fell below one pixel. Maximum feature excursion was measured as the maximum distance of all feature points to the principal point (image center) over the entire servoing process. This metric reveals problems with feature visibility. Maximum camera excursion was defined as the maximal distance of the camera from the goal position. This is not very interesting in the context of shortest-path servoing, where the maximum excursion should only depend on the starting position. Maximum camera rotation was tracked by transforming the rotations into axis-angle form and finding the maximum of the rotation angle. Only the relevant metrics are presented for each servoing task.

The servoing target consisting of six points can be seen in Fig. 1. Compared to Gans et al.[8], we have decided not to use a fully planar target but instead to displace two of the feature points by a small amount to avoid the singularity occurring when the points become collinear in the image. In addition, we use the full pose estimation instead of a homography to evaluate the method by Deguchi, because when the points become almost planar, the homography based estimation approaches a singularity.

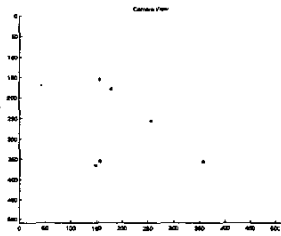


Fig. 1. Servoing target.

Only the new position-based approach presented in Sec. IV is experimentally evaluated, since the hybrid approach has exactly the same control function and the only difference remains in the pose estimation method. We do not want to evaluate different methods for pose estimation, because that would divert our focus from visual servo control. We chose to use the linear pose estimation algorithm introduced by Fiore [14] due to the suitability of non-iterative approaches for visual servoing. All simulation experiments were performed in Simulink with a variable time-step.

A. Optical axis rotation

Optical axis rotation is known to be problematic task for IBVS. In addition, other methods which use image Jacobian to control rotation, such as Deguchi's method, are prone to the singularity of the Jacobian.

Figure 2 shows the time of convergence and maximum feature excursion for optical axis rotation. Deguchi's method never converges for 180° rotation due to the

singularity problem, and the time of convergence increases significantly when the rotation angle increases. The method presented here performs identically to PBVS, the jagged curve resulting from variable time-step simulation with a maximum step size of 0.1 seconds. Both the new method and PBVS rotate the camera only around the optical axis. Deguchi's method rotates also other axes because the servoing target is not symmetric. Thus, it can be expected to make unnecessary rotations in this case.

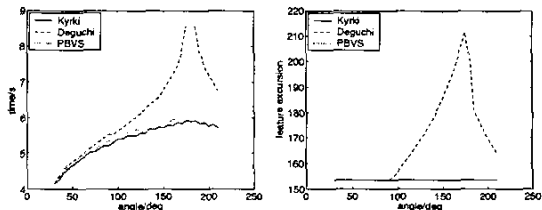


Fig. 2. Optical axis rotation.

B. Optical axis translation

The initial positions for translation along the optical axis range from 1 m retreated to 1 m advanced. Optical axis translation is dependent on depth estimation which can be critical to some methods. Here, all three methods have essentially the same control law which can be seen from the results in Fig. 3. The differences are again a result of the variable time-step simulation.

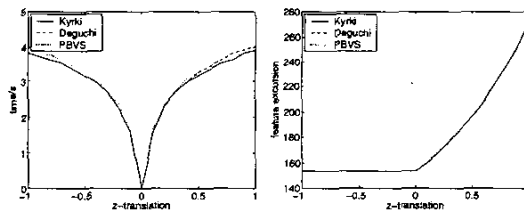


Fig. 3. Optical axis translation.

C. y-axis rotation

The y -axis rotation represents rotation of the camera perpendicular to optical axis. Because even small rotations cause the features to be lost from the field of view, we follow Gans et al. and allow an infinite image plane in this test. Here, Fig. 4 presents all metrics because some results differ from those given in [8]. The time to convergence seems to be approximately linear to the amount of rotation for all methods. The maximum feature excursion is also identical. These results are consistent with those given by Gans et al. and the differences are explained by the different servoing target. Maximum camera translation is also close to zero as all methods use position based control for translation which results in zero translational velocity. The difference from results published by Gans et al. is in

the measurement of the maximum camera rotation. Our experiments indicate that with all methods the maximal rotation angle is identical to the initial rotation. Their graph on rotation has a maximum of approximately 18° at 30° degree rotation around y -axis. All of our experiments show a linear relationship between the initial and maximal rotations. Otherwise our results agree with theirs.

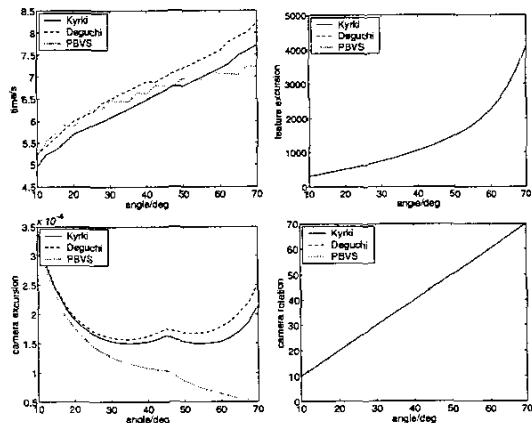


Fig. 4. y -axis rotation.

D. Feature plane rotation

In the feature plane rotation the points are rotated about an axis perpendicular to the optical axis and lying in the feature point plane. Thus, both rotation and translation need to be controlled. To compare the methods fairly, we used the true depths in the image Jacobian for Deguchi's method. Figure 5 presents the time to convergence and the feature excursion. In this task, the method presented in this paper and Deguchi's method have similar dependence between rotation angle and time to convergence. For PBVS, the time increases slightly faster. For small rotations the feature excursion is identical, but the differences can be seen as the point where visual features begin to divert from the origin. For PBVS this happens at approximately 30° , for our method at 45° and for Deguchi's method at 55° . However, for all three methods the features remain in the field of the view of the camera throughout the servoing.

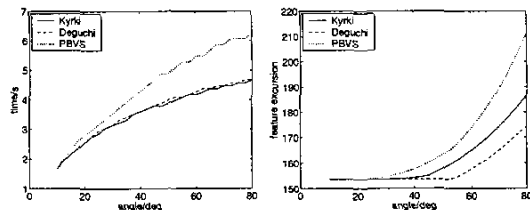


Fig. 5. Feature plane rotation.

E. General motion

The general motion task combines the rotation around optical axis with the feature plane rotation. The starting pose is determined from the goal pose as follows: First, the pose is rotated by a specified amount around the optical axis. Then, the pose is rotated by the same amount around a perpendicular axis on the feature point plane. Time to convergence and feature point excursion are shown in Fig. 6. Time to convergence grows slightly faster for PBVS than for the other two methods. After 50° the time increases slightly faster also for Deguchi's method compared to the proposed one. However, the difference is almost marginal. The real difference of the methods can be seen in the feature excursion. With PBVS, the features soon begin to divert far from the optical center. After 50° some of the features leave the camera field of view altogether. However, in the simulation we have allowed an infinite image area to better demonstrate this phenomenon. Both our and Deguchi's methods control the visibility efficiently with only marginal difference in performance. The image trajectories of the methods are shown in Fig. 7. Again, the trajectories of our and Deguchi's methods are similar, and the visibility problem of PBVS can be clearly seen.

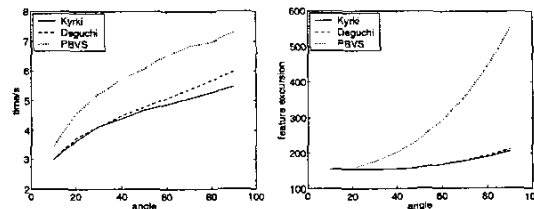


Fig. 6. General motion.

VII. DISCUSSION

In this paper, we have proposed a shortest-path servoing strategy to visual servo control. The algorithm is convergent in the whole task space except in the degenerate case when the servoing target lies in the camera image plane.

Simulation experiments were performed to compare the method to previously introduced shortest-path methods. The experiment on optical-axis rotation revealed that the method does not suffer from convergence problems with large optical axis rotations, unlike the method by Deguchi, which uses several image points to control the rotation. The experiment on y -axis rotation, where each of the methods has a different control function, shows that the performance is comparable between all methods. In feature plane rotation, the partitioned methods converge faster than PBVS. However, for all methods the image features remain in the field of view throughout the servoing, and all are therefore able to converge. In the general motion experiment, PBVS exhibits the problem of image features leaving the field of view.

Estimation of planar homography requires that feature points lie in a plane. However, the hybrid method could

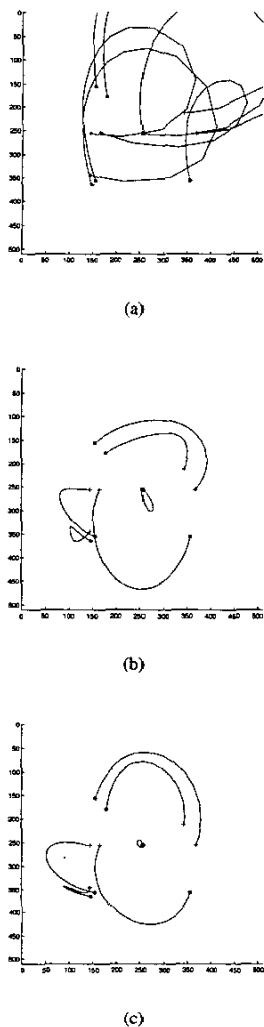


Fig. 7. Trajectories for general motion: a) PBVS, b) Deguchi, c) Kyrki et al.

also use essential matrix decomposition instead of homography, as proposed by Deguchi [7] and Malis [15]. The use of homography is likely to give more stable results, especially near the convergence of the system [15].

The straight line shortest-path servoing presented avoids reaching the joint limits of a robot in most servoing tasks. However, while guaranteeing that the object origin is always visible, the camera can get too close to the object so that point features are lost. Possible solutions include the use of switching between servoing strategies or repulsive potential field in the image or 3D. However, the straight line trajectory can not be attained with these approaches.

Besides the Cartesian path length, other metrics could be used to evaluate the motion of a robot, for example, time of

convergence or the energy consumption during the servoing task. However, these metrics are strongly dependent on the mechanical structure of a robot and its dynamic properties, and universal approaches seem thus difficult to construct. In some cases the shortest-path is not sound such in the case where the starting point is on the opposite side of the target compared to the goal, and the shortest path would pass through the target. However, it is unlikely that every imaginable task can be solved by any one servoing strategy without shortcomings.

ACKNOWLEDGMENT

V. Kyrki was supported by a grant from the Academy of Finland. The support is gratefully acknowledged.

REFERENCES

- [1] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 651–670, Oct. 1996.
- [2] W. J. Wilson, C. C. W. Hulls, and G. S. Bell, "Relative end-effector control using cartesian position based visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 684–696, Oct. 1996.
- [3] F. Chaumette, "Potential problems of stability and convergence in image-based and position-based visual servoing," in *The Confluence of Vision and Control*, ser. Lecture Notes in Control and Information Sciences, no. 237. Springer-Verlag, 1998, pp. 66–78.
- [4] E. Malis and P. Rives, "Robustness of image-based visual servoing with respect to depth distribution errors," *IEEE International Conference on Robotics and Automation*, pp. 1056–1061, Sept. 2003.
- [5] F. Chaumette, E. Malis, and S. Boudet, "2D 1/2 visual servoing with respect to a planar object," in *Proc. Workshop on New Trends in Image-Based Robot Servicing*, 1997, pp. 44–52.
- [6] E. Malis, F. Chaumette, and S. Boudet, "2-1/2-D visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 238–250, Apr. 1999.
- [7] K. Deguchi, "Optimal motion control for image-based visual servoing by decoupling translation and rotation," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Victoria, B.C., Canada, Oct. 1998, pp. 705–711.
- [8] N. R. Gans, S. A. Hutchinson, and P. I. Corke, "Performance tests for visual servo control systems with application to partitioned approaches to visual servo control," *The International Journal of Robotics Research*, vol. 22, no. 10–11, pp. 955–981, Oct.–Nov. 2003.
- [9] P. I. Corke and S. A. Hutchinson, "A new partitioned approach to image-based visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 4, pp. 507–515, Aug. 2001.
- [10] Y. Mezouar and F. Chaumette, "Path planning for robust image-based control," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 4, pp. 534–549, Aug. 2002.
- [11] N. R. Gans and S. A. Hutchinson, "An asymptotically stable switched system visual controller for eye in hand robots," in *Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, Las Vegas, Nevada, Oct. 2003.
- [12] O. D. Faugeras and F. Lustran, "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.
- [13] E. Malis, F. Chaumette, and S. Boudet, "Positioning a coarse-calibrated camera with respect to an unknown object by 2D 1/2 visual servoing," in *IEEE International Conference on Robotics and Automation*, vol. 2, Leuven, Belgium, May 1998, pp. 1352–1359.
- [14] P. D. Fiore, "Efficient linear solution of exterior orientation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 140–148, Feb. 2001.
- [15] E. Malis and F. Chaumette, "2 1/2 D visual servoing with respect to unknown objects through a new estimation scheme of camera displacement," *International Journal of Computer Vision*, vol. 37, no. 1, pp. 79–97, 2000.