Straight Line Path-Planning in Visual Servoing

Graziano Chesi
Department of Electrical and Electronic Engineering,
University of Hong Kong,
Pokfulam Road, Hong Kong
e-mail: chesi@eee.hku.hk

Domenico Prattichizzo
e-mail: prattichizzo@dii.unisi.it

Antonio Vicino
e-mail: vicino@dii.unisi.it

Department of Information Engineering,
University of Siena,
Via Roma 56,
53100 Siena, Italy

This paper deals with visual servoing for 6-degree-of-freedom robot manipulators, and considers the problem of establishing whether and how it is possible to reach the desired location while keeping all features in the field of view and following a straight line in the Euclidean space. A path-planning technique based on a parametrization of the camera path through polynomials is proposed, which overcomes existing methods dealing with this problem. The generated image trajectory can be tracked by using an image-based visual servoing controller.
[DOI: 10.1115/1.2745882]

Keywords: visual servoing, point correspondences, straight line, path-planning

1 Introduction

The "teaching-by-showing" approach has received the attention of several researchers in recent years. It consists of teaching the desired location for an eye-in-hand camera by showing the view of some reference features in such a location. The camera is then moved to another location and must be steered to the goal location by exploiting the current and desired view of the features. Several methods to deal with this task have been proposed, such as position-based visual servoing, image-based visual servoing (IBVS), and 2-1/2 D visual servoing (see, for example, Ref. [1], and references therein). Other methods include navigation functions [2], invariance with respect to the intrinsic parameters [3], and generation of circular-like trajectories [4]. These methods aim mainly to guarantee convergence, to keep features in the field of view, and to confer robustness against uncertainties.

Another problem of theoretical and practical interest is to establish whether and how the camera may reach the desired location while keeping all features in the field of view and following a straight line in the Euclidean space. Some of the existing methods can approach this problem, but only in quite restrictive conditions. For example, in the path-planning method [5] the image trajectory is built by initially following a geodesic in SE(3), which restricts the set of trajectories whose projection on the Euclidean space is a straight line. In Ref. [6], a method to follow a straight line is proposed based on the control of two rotation axes to make a reference point moving on a straight line in the image; however, the other points may get move out of the field of view. Another method has been proposed in Ref. [7] where the image trajectory is generated by following an a priori selected screw motion, which represents only one of the sought trajectories.

In this paper the problem is dealt with for 6-degree-of-freedom (DOF) robot manipulators by rationally parametrizing the rotation path through the Cayley parameter. This allows us to parametrize all rotation paths and derive a polynomial formulation of the arising optimization that can be more easily solved. Indeed, the distance of the image trajectory from the screen boundary is simply calculated by finding the roots of some one-variable polynomials. The image trajectory is generated through a scaled Euclidean reconstruction of the observed object which can be easily estimated from point correspondences in the initial and desired camera views. The camera is finally steered to the desired location by tracking the computed image trajectory with an IBVS controller.

2 Problem Formulation and Preliminaries

Let $\mathbf{I}_n$ denote the identity matrix $n\times n$, $\mathbf{0}_n$ the null vector $n\times 1$, $\mathbf{1}_n$ the vector $n\times 1$ with all elements equal to 1, $\mathbf{e}_i$ the ith column of $\mathbf{I}_3$, $\mathbf{v}_{|i}$ the ith component of vector $\mathbf{v}$, $[\mathbf{v}]_\times$ the skew-symmetric matrix of $\mathbf{v}\in\mathbb{R}^3$, SO(3) the set of the rotation matrices in $\mathbb{R}^{3\times 3}$, and $\mathbf{A}\in\mathbb{R}^{3\times 3}$ the upper triangular intrinsic parameters matrix. Moreover, let $\mathcal{F}$and $\mathcal{F}^*$ denote the initial and desired camera frames, respectively, $\mathbf{R}\in SO(3)$ and $\mathbf{t}\in\mathbb{R}^3$ the rotation and translation of $\mathcal{F}^*$, respectively, with respect to $\mathcal{F}$ expressed in $\mathcal{F}$, and $\mathbf{p}_i=[x_i,y_i,1]^T\in\mathbb{R}^3$ and $\mathbf{p}_i^*=[x_i^*,y_i^*,1]^T\in\mathbb{R}^3$ the projection of the ith point on $\mathcal{F}$ and $\mathcal{F}^*$ in pixel homogeneous coordinates.

Let us suppose that a set $\mathcal{P}=\{(\mathbf{p}_i,\mathbf{p}_i^*),i=1,\ldots,N\}$ of N point correspondences is available. The problem dealt with is to establish whether and how it is possible to reach the desired location while keeping all points in the field of view and following a straight line in the Euclidean space.

From $\mathcal{P}$ and $\mathbf{A}$, one can compute the camera pose $\mathbf{R}$ and $\mathbf{t}_n=\mathbf{t}/\|\mathbf{t}\|$ (for pure rotation motion, i.e., $\mathbf{t}=\mathbf{0}_3$, we define $\mathbf{t}_n=\mathbf{0}_3$). This can be done through the essential matrix algorithm or the homography matrix algorithm relative to a virtual plane in the case of non-coplanar features supposing $N\geq 8$ (see Refs. [8,9] for details). One can then compute a scaled Euclidean reconstruction $\mathcal{Q}=\{\mathbf{q}_i\in\mathbb{R}^3,i=1,\ldots,N\}$ of the object by solving

$$\alpha_i\mathbf{p}_i=\mathbf{A}\mathbf{q}_i,\quad \alpha_i^*\mathbf{p}_i^*=\mathbf{A}\hat{\mathbf{R}}(\mathbf{q}_i-\mathbf{t}_n) \tag{1}$$

where $\alpha_i,\alpha_i^*\in\mathbb{R}$ are the scaled depths. By eliminating α_i and α_i^*, (1) becomes affine linear in $\mathbf{q}_i$ and can be rewritten as

$$[\mathbf{q}_i^T,1][\mathbf{U}_i,\mathbf{U}_i^*]=\mathbf{0}_4^T \tag{2}$$

where $\mathbf{U}_i,\mathbf{U}_i^*\in\mathbb{R}^{4\times 2}$ are suitable constant matrices.

The proposed approach exploits the representation of rotation matrices through the Cayley parameter (see, for example, Ref. [10]). Let us define

$$\mathbf{\Gamma}(\mathbf{a})=(\mathbf{I}_3-[\mathbf{a}]_\times)^{-1}(\mathbf{I}_3+[\mathbf{a}]_\times) \tag{3}$$

It turns out that $\mathbf{\Gamma}(\mathbf{a})\in SO(3)$ for all $\mathbf{a}\in\mathbb{R}^3$. Moreover, for all $\mathbf{R}\in SO(3)$ there exists $\mathbf{a}$, called *Cayley parameter* of $\mathbf{R}$, such that $\mathbf{R}=\mathbf{\Gamma}(\mathbf{a})$. This parameter can be found according to $\|\mathbf{a}\|=\tan\theta/2$ and $\mathbf{a}\|\mathbf{a}\|^{-1}=\mathbf{u}$, where $\theta\in[0,\pi]$ and $\mathbf{u}\in\mathbb{R}^3$, $\|\mathbf{u}\|=1$, are, respectively, the rotation angle and axis in the exponential coordinates of $\mathbf{R}$; i.e., $\mathbf{R}=e^{[\theta\mathbf{u}]_\times}$.

3 Path Planning

Let $\hat{\mathbf{A}}$ and $\hat{\mathcal{P}}=\{(\hat{\mathbf{p}}_i,\hat{\mathbf{p}}_i^*),i=1,\ldots,N\}$ be the estimates of $\mathbf{A}$ and $\mathcal{P}$ affected by calibration errors and image noise, and let $\hat{\mathbf{R}}$, $\hat{\mathbf{t}}_n$, and $\hat{\mathbf{q}}_i$ be the estimates of the camera pose and scaled Euclidean re-

Contributed by the Dynamic Systems, Measurement, and Control Division of ASME for publication in the JOURNAL OF DYNAMIC SYSTEMS, MEASUREMENT, AND CONTROL. Manuscript received October 7, 2005; final manuscript received February 5, 2007. Review conducted by George Chiu.

construction. Let $w \in [0,1]$ be the path abscise, with $w=0$ and $w=1$ denoting, respectively, the initial and desired location. The ith image point $\mathbf{p}_i(w)$ along the path can be obtained as

$$\alpha_i(w)\mathbf{p}_i(w) = \hat{\mathbf{A}}\mathbf{R}_c(w)[\hat{\mathbf{q}}_i - \mathbf{t}_c(w)] \tag{4}$$

where $\alpha_i(w)$, $\mathbf{R}_c(w)$, and $\mathbf{t}_c(w)$ are the scaled depth and camera pose along the path. By imposing the boundary constraints, one has

$$\mathbf{p}_i(0) = \hat{\mathbf{p}}_i \quad \mathbf{p}_i(1) = \hat{\mathbf{p}}_i^*$$

$$\mathbf{R}_c(0) = \mathbf{I}_3 \quad \mathbf{R}_c(1) = \hat{\mathbf{R}}$$

$$\mathbf{t}_c(0) = \mathbf{0}_3 \quad \mathbf{t}_c(1) = \hat{\mathbf{t}}_n \tag{5}$$

In order to follow a straight line in the Euclidean space, we can select $\mathbf{t}_c(w)=w\hat{\mathbf{t}}_n$. Let us observe that the first constraint in (5) is satisfied if and only if $[\hat{\mathbf{q}}_i^T,1]^T[\hat{\mathbf{U}}_i,\hat{\mathbf{U}}_i^*]=\mathbf{0}_4^T$ admits a solution for $\hat{\mathbf{q}}_i$, where $\hat{\mathbf{U}}_i,\hat{\mathbf{U}}_i^*$ are the matrices $\mathbf{U}_i,\mathbf{U}_i^*$ affected by the calibration errors and image noise. This solution does not exist in the general case because the number of constraints is larger than the number of free variables. This means that the image trajectory $\mathbf{p}_i(w)$ could not start in $\hat{\mathbf{p}}_i$ and could not end in $\hat{\mathbf{p}}_i^*$. In order to overcome this problem, we calculate the estimate $\hat{\mathbf{q}}_i$ by ensuring the right final point of the trajectory (since this is essential to ensure convergence to the desired location) and using the remaining degrees of freedom to minimize the error at the initial point:

$$\min_{\hat{\mathbf{q}}_i} \|[\hat{\mathbf{q}}_i^T,1]^T\hat{\mathbf{U}}_i\|^2 \text{ s.t. } [\hat{\mathbf{q}}_i^T,1]^T\hat{\mathbf{U}}_i^* = \mathbf{0}_4^T \tag{6}$$

The optimal value of $\hat{\mathbf{q}}_i$ in (6) can be easily obtained through trivial calculations. The next step to parametrize the image trajectory is to satisfy the visibility constraint $x_m < \mathbf{p}_{i|1}(w) < x_M$ and $y_m < \mathbf{p}_{i|2}(w) < y_M$ for all $w \in [0,1]$ where x_m, x_M, y_m, and y_M are the screen limits. Hence, let μ be the distance from the image trajectory to the screen boundary defined as

$$\mu = \min_{w\in[0,1],i=1,\ldots,n} \mu_i(w) \tag{7}$$

$$\mu_i(w) = \min[\mathbf{p}_{i|1}(w) - x_m, x_M - \mathbf{p}_{i|1}(w), \mathbf{p}_{i|2}(w) - y_m, y_M - \mathbf{p}_{i|2}(w)] \tag{8}$$

Clearly, a path satisfies the visibility constraint if and only if $\mu > 0$. To find such a path, one has to maximize μ over all possible rotational trajectories that satisfy the second constraint in (5). Computing μ may be a difficult task depending on the dependency of $\mathbf{R}_c(w)$ from w. One way to simplify this task is to select a rational parametrization for $\mathbf{R}_c(w)$. This can be done by exploiting the Cayley parameters, in particular by rationally parametrizing the Cayley parameter $\mathbf{a}(w)$ of $\mathbf{R}_c(w)$. Hence, let us express $\mathbf{a}(w)$ as

$$\mathbf{a}(w) = \frac{\mathbf{n}(w)}{1+\gamma w} \quad \mathbf{n}(w) = \mathbf{N}[w^\delta, w^{\delta-1}, \ldots, 1]^T$$

$$\gamma = \begin{cases} 0 & \text{if } \hat{\mathbf{R}} \notin \mathbf{\Pi} \\ -1 & \text{otherwise} \end{cases} \tag{9}$$

$$\mathbf{\Pi} = \{\mathbf{R} \in \mathrm{SO}(3) : \mathbf{R} = e^{[\theta\mathbf{u}]_\times}, \theta = \pi\} \tag{10}$$

where δ is the degree of the numerator and $\mathbf{N} \in \mathbb{R}^{3\times(\delta+1)}$ is a coefficient matrix. The second constraint in (5) imposes

$$\mathbf{n}(0) = \mathbf{0}_3 \quad \mathbf{n}(1) = \begin{cases} \hat{\mathbf{a}} & \text{if } \hat{\mathbf{R}} \notin \mathbf{\Pi} \\ \hat{\mathbf{a}}\|\hat{\mathbf{a}}\|^{-1} & \text{otherwise} \end{cases} \tag{11}$$

where $\hat{\mathbf{a}}$ is the Cayley parameter of $\hat{\mathbf{R}}$. Hence, $\mathbf{N}$ can be parametrized as

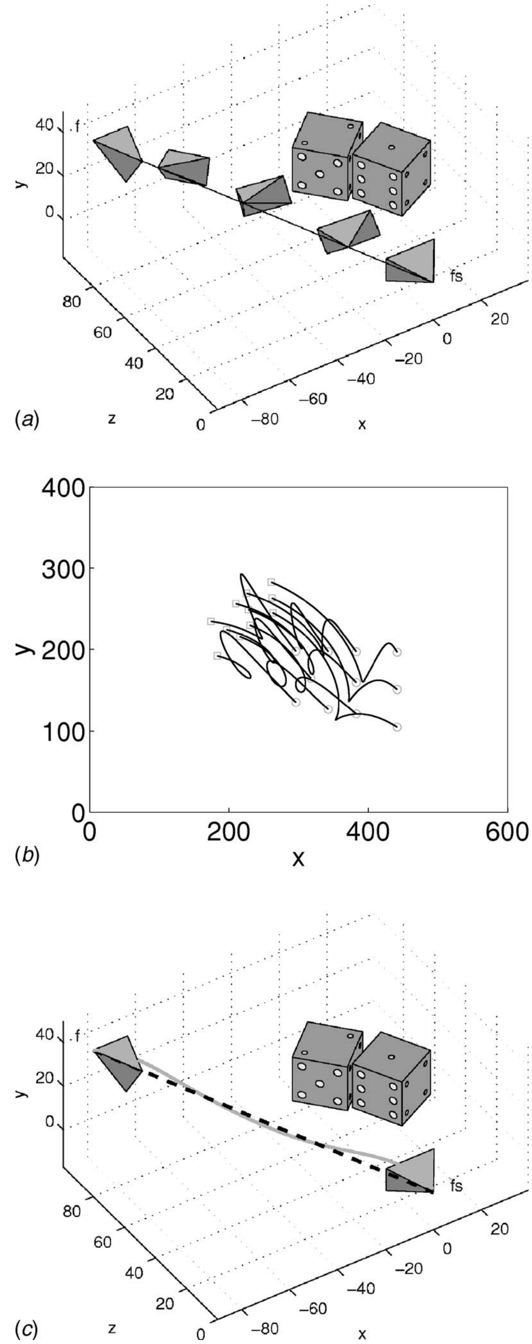

Fig. 1 (*a*) Ideal case (scene); (*b*) ideal case (camera view); (*c*) presence of uncertainties case (scene)

$$\mathbf{N} = [\mathbf{C}, \mathbf{n}(1) - \mathbf{C}\mathbf{1}_{\delta-1}, \mathbf{0}_3] \tag{12}$$

where $\mathbf{C} \in \mathbb{R}^{3\times(\delta-1)}$ is a free parameter. In order to take into account the visibility constraints, let us define, for $i=1,\ldots,n$ and $j=1,2,3$, the polynomials of degree $2\delta+1$:

$$b_{i,j}(w) = \mathbf{e}_j^T\hat{\mathbf{A}}\mathbf{\Omega}[\mathbf{n}(w), 1+\gamma w](\hat{\mathbf{q}}_i - w\mathbf{t}_n) \tag{13}$$

$$\mathbf{\Omega}(\mathbf{n},h) = (h^2 + \|\mathbf{n}\|^2)\mathbf{\Gamma}(h^{-1}\mathbf{n}) \tag{14}$$

($\mathbf{\Omega}(\mathbf{n},h)$ is, in fact, a quadratic function of $\mathbf{n}$ and h). The ith image point can be expressed as $\mathbf{p}_i(w) = [b_{i,1}(w), b_{i,2}(w), b_{i,3}(w)]^T b_{i,3}(w)^{-1}$. The points lie in front of the camera if $d > 0$, where

$$d = \min_{w \in [0,1], i=1,\ldots,n} b_{i,3}(w) \tag{15}$$

The distance from the image trajectory to the screen boundary is

$$\mu_\delta = \begin{cases} -\infty & \text{if } d \leq 0 \\ \sup \varepsilon \text{ s.t. } (17) & \text{otherwise} \end{cases} \tag{16}$$

$$\left.\begin{array}{l} b_{i,1}(w) - (x_m + \varepsilon) b_{i,3}(w) > 0 \\ -b_{i,1}(w) + (x_M - \varepsilon) b_{i,3}(w) > 0 \\ b_{i,2}(w) - (y_m + \varepsilon) b_{i,3}(w) > 0 \\ -b_{i,2}(w) + (y_M - \varepsilon) b_{i,3}(w) > 0 \end{array}\right. \forall \;\; w \in [0,1] \;\; \forall \;\; i = 1, \ldots, n \tag{17}$$

The maximum distance achievable with a Cayley parameter of degree δ is hence

$$\mu_\delta^* = \sup_{\mathbf{C}} \mu_\delta \tag{18}$$

The quantity d can be simply found by evaluating each $b_{i,3}(w)$ at the points where its derivative vanishes in $[0,1]$ and extremes $w=0$ and $w=1$. Moreover, it is easy to see that also μ_δ can be found through a similar investigation on suitable rational functions of w.

4 Example

Let us consider the situation depicted in Fig. 1(a), where the set of points in the "5" and "6" faces of a pair of dice represents the observed object. The intrinsic parameters matrix of the camera is $\mathbf{A} = [400, 0, 300; 0, 400, 200; 0, 0, 1]$ pixels. Let us consider first the ideal case, i.e., absence of calibration errors and image noise. Through the essential matrix algorithm we estimate the camera pose, and we then compute the scaled Euclidean reconstruction in (6) and solve the problem (18). For $\delta = 3$ we find $\hat{\mu}_3^* = 104.4$ pixels corresponding to the image trajectory shown in Fig. 1(b) and to the camera trajectory shown in Fig. 1(a) (straight line).

Let us consider now a real case, where the bad estimate of $\mathbf{A}$ is $\hat{\mathbf{A}} = [365, 0, 330; 0, 420, 185; 0, 0, 1]$ and the image noise is generated through a random shift of each image point in a square with side equal to 1 pixel. The image trajectory is planned analogously to the ideal case, and the found trajectory is tracked by using the IBVS controller proposed in Ref. [5]. Figure 1(c) shows the final camera trajectory. Clearly, the presence of uncertainties in the planning and tracking stages causes a deviation of the camera trajectory from the planned one, and a similar deviation is caused in real applications by the non-perfect robot controller. However, let us observe that the deviation caused by image noise and calibration errors is quite minor (the trajectory length is 133.8 cm in the real case and 130.4 cm in the ideal case). Moreover, let us observe that existing methods are often unable to follow a straight line even in ideal conditions, as explained in Sec. 1.

5 Conclusion

This paper proposes for 6-DOF robot manipulators a path-planning technique for keeping the features in the field of view while following a straight line in the Euclidean space. In particular, a parametrization through polynomials of all camera paths whose projection on the Euclidean space is a straight line is introduced. The contribution with respect to existing works is to provide a non-restrictive technique to plan the camera path.

References

[1] Hashimoto, K., 2003, "A Review on Vision-Based Control of Robot Manipulators," Adv. Rob., **17**(10), pp. 969–991.

[2] Cowan, N. J., Weingarten, J. D., and Koditschek, D. E., 2002, "Visual Servoing via Navigation Functions," IEEE Trans. Rob. Autom., **18**(4), pp. 521–533.

[3] Malis, E., 2004, "Visual Servoing Invariant to Changes in Camera-Intrinsic Parameters," IEEE Trans. Rob. Autom., **20**(1), pp. 72–81.

[4] Chesi, G., and Vicino, A., 2004, "Visual Servoing for Large Camera Displacements," IEEE Trans. Rob. Autom., **20**(4), pp. 724–735.

[5] Mezouar, Y., and Chaumette, F., 2002, "Path Planning for Robust Image-Based Control," IEEE Trans. Rob. Autom., **18**(4), pp. 534–549.

[6] Kyrki, V., Kragic, D., and Christensen, H. I., 2004, "Measurement Errors in Visual Servoing," in *Proc. IEEE Int. Conf. on Robotics and Automation*, New Orleans, Louisiana, pp. 1861–1867.

[7] Park, J. S., and Chung, M. J., 2003, "Path Planning With Uncalibrated Stereo Rig for Image-Based Visual Servoing Under Large Pose Discrepancy," IEEE Trans. Rob. Autom., **19**(2), pp. 250–258.

[8] Faugeras, O., and Luong, Q.-T., 2001, *The Geometry of Multiple Images*, MIT Press, Cambridge, MA.

[9] Malis, E., and Chaumette, F., 2004, "2 1/2 D Visual Servoing With Respect to Unknown Objects Through a New Estimation Scheme of Camera Displacement," Int. J. Comput. Vis., **37**(1), pp. 79–97.

[10] Murray, R. M., Li, Z., and Sastry, S. S., 1994, *A Mathematical Introduction to Robotic Manipulation.* CRC Press, Boca Raton, FL.