2D VISUAL SERVOING ON COMPLEX SHAPES BASED ON A POLAR DESCRIPTION

Christophe Collewet [*] **François Chaumette** [**]
Laurence Wallian [*]

[*] *Cemagref, UR Technologie des équipements agro-alimentaires,*
17 Avenue de Cucillé, 35044 Rennes Cedex, France
[**] *IRISA / INRIA, Campus Universitaire de Beaulieu,*
35042 Rennes Cedex, France

Abstract: This paper deals with the way to achieve robotic positioning tasks by 2D visual servoing. We consider the case when the considered objects have a complex and unknown shape. First, we are interested in the computation of an analytical expression of the interaction matrix with regards to the visual features extracted from the image contour of the observed object. We illustrate this way to proceed thanks to a polar description of the contour. Experimental results validate the proposed algorithm. In particular, the robustness of the control law is tested with regards to a coarse calibrated system, to an approximation of the depth of the object, and to partial occlusion. *Copyright © 2000 IFAC*

Keywords: 2D visual servoing, interaction matrix, complex visual features, polar description of a contour

1. INTRODUCTION

Image-based visual servoing (Hutchinson *et al.*, 1996) depends by nature on the choice of the visual features. Thus, the robotic tasks to be performed depend on the observed object in the sense that the interaction matrix, associated with the visual features extracted from the image, can be obtained or not. This matrix is essential for such an approach. Most often, points of interest or simple geometric primitives (*e.g.* lines, circles, cylinders, spheres) are used (Feddema *et al.*, 1989; Espiau *et al.*, 1992). In (Espiau *et al.*, 1992), the authors proposed a general method to obtain the interaction matrix when the considered object can be represented by a parametric equation. However, this method cannot be used when no analytic representation of the visual features is provided, which is the case in most real cases.

Several works deal with visual features able to describe an image overall (Bien *et al.*, 1993; Deguchi and Noguchi, 1996; Wells *et al.*, 1996; Nayar *et al.*, 1996). However these approaches are all based on a learning step. New learning phases are therefore necessary when the robot has to deal with a new object. Such phases may be heavy (Deguchi and Noguchi, 1996) even when the nature of the visual features are unchanged. In fact, the way such features change cannot be expressed analytically when the robot moves. More, it is impossible to give a proof of the stability of the system. Thus it is in our interest, from a theorical and practical point of view, to elaborate control laws based on an analytic approach. This is why a contour approach seems to be interesting. Such an approach has been proposed in (Drummond and Cipolla, 1999; Colombo and Allotta, 1999). The visual features used in the control scheme are the parameters of an affine deformation of the contour from the current to the desired images. Unfortunately, this approach leads to a singularity when the object and the image plane are parallel. Besides, the matching between the two contours may be complex. We will see that these problems do not occur in the method proposed here.

After having recalled in Section 2 some fundamentals on 2D visual servoing, we show in Section 3 a new approach of the interaction matrix computation based on a polar description of the image of the object. Next, in Section 4, we describe how to extract the visual features from the image in practice. In Section 5, experimental results validate our approach, positioning tasks with regards to objects with complex shape are achieved. Finally, a concluding section summarizes the main results.

2. IMAGE-BASED VISUAL SERVOING

In image-based visual servoing, the control scheme is performed on the basis of visual features extracted directly from the image: from a m-dimensional vector $\underline{s}$ describing the current visual features the goal is to move the robot so that $\underline{s} = \underline{s}^*$ where $\underline{s}^*$ describes the features when the robot is at the desired position. Such an approach is based on the relationship between the camera velocity T_c and the visual features velocity $\dot{\underline{s}}$. This relationship is described by a matrix called the interaction matrix (or image jacobian):

$$\dot{\underline{s}} = L_{\underline{s}}^T T_c \qquad (1)$$

where $T_c = (\underline{V}^T, \underline{\Omega}^T)^T$ with $\underline{V} = (v_x, v_y, v_z)^T$ and $\underline{\Omega} = (\omega_x, \omega_y, \omega_z)^T$ are the translational and rotational components of T_c respectively.

A vision-based task $\underline{e}$ can thus be defined by:

$$\underline{e}(\underline{r}, t) = C\,(\underline{s}(\underline{r}, t) - \underline{s}^*) \qquad (2)$$

where C is a combination matrix taking into account cases where m is different from n (with n the number of robot d.o.f.). It has to be chosen so that $CL_{\underline{s}}^T$ is full rank and, if $L_{\underline{s}}^T$ is full rank, it can be defined as:

$$C = \widehat{L_{\underline{s}}^{T+}} \qquad (3)$$

where $\widehat{L_{\underline{s}}^T}$ is a model or an approximation of $L_{\underline{s}}^T$.

In case of a motionless object, the camera velocity can be obtained by:

$$T_c = -\lambda \underline{e} \qquad (4)$$

This relationship ensures an exponential decrease of $\underline{e}$ ($\dot{\underline{e}} = -\lambda \underline{e}$) when $\widehat{L_{\underline{s}}^T} = L_{\underline{s}}^T$. Otherwise it occurs when the condition $\widehat{L_{\underline{s}}^T}^{+} L_{\underline{s}}^T > 0$ is respected (Espiau *et al.*, 1992).

After having recalled this fundamentals, we first describe a way to obtain an analytic representation of the interaction matrix by using a polar description of the object contours in the image.

3. COMPUTATION OF THE INTERACTION MATRIX

Let us consider a point $m = (x, y, z)^T$ expressed in the camera coordinate system (with z axis $\equiv$ optical axis). It projects on the image plane in $M = (X, Y)^T$ that can be written in a polar description by:

$$\underline{OM} = (X_c + \rho\cos\theta)\underline{u}_x + (Y_c + \rho\sin\theta)\underline{u}_y \qquad (5)$$

where (X_c, Y_c) are the coordinates of the centroid in the image.

This leads to the velocity of M:

$$\begin{cases} \dot{X} = \dot{\rho}\cos\theta - \rho\dot{\theta}\sin\theta + \dot{X}_c \\ \dot{Y} = \dot{\rho}\sin\theta + \rho\dot{\theta}\cos\theta + \dot{Y}_c \end{cases} \qquad (6)$$

The radius ρ being periodic, we can perform a Fourier expansion:

$$\rho(\theta) = a_0 + \sum_{k=1}^{k=h} a_k \cos k\theta + b_k \sin k\theta \qquad (7)$$

where h is the number of harmonics taken into account.

By denoting $\underline{P} = (a_0, \cdots, a_h, b_1, \cdots, b_h)^T$ and eliminating $\dot{\theta}$, (6) becomes:

$$\dot{X} - \alpha\dot{Y} = \dot{X}_c - \alpha\dot{Y}_c + \beta\frac{\partial\rho}{\partial\underline{P}}\dot{\underline{P}} \qquad (8)$$

where we have:

$$\alpha = \frac{\dfrac{\partial\rho}{\partial\theta}\cos\theta - \rho\sin\theta}{\dfrac{\partial\rho}{\partial\theta}\sin\theta + \rho\cos\theta} \text{ and } \beta = \cos\theta - \alpha\sin\theta$$

Then, we choose for the visual features the vector $\underline{Q} = (X_c, Y_c, \underline{P}^T)$. Therefore, relation (8) can be written as:

$$\dot{X} - \alpha\dot{Y} = \left(1, -\alpha, \beta\frac{\partial\rho}{\partial\underline{P}}\right)\dot{\underline{Q}}^T \qquad (9)$$

If we choose $m = 2h + 3$ independent pairs (ρ, θ) and inverse the system obtained, we have:

$$\dot{\underline{Q}}^T = M \begin{pmatrix} \dot{X}_1 - \alpha_1\dot{Y}_1 \\ \vdots \\ \dot{X}_m - \alpha_m\dot{Y}_m \end{pmatrix} \qquad (10)$$

in which:

$$M = \begin{pmatrix} 1 & -\alpha_1 & \beta_1 \left.\dfrac{\partial\rho}{\partial\underline{P}}\right|_{\rho_1,\theta_1} \\ \vdots & \vdots & \vdots \\ 1 & -\alpha_m & \beta_m \left.\dfrac{\partial\rho}{\partial\underline{P}}\right|_{\rho_m,\theta_m} \end{pmatrix}^{-1} \qquad (11)$$

Thereafter, by substituting $\dot{X}$ and $\dot{Y}$ where $\dot{X} = L_{\dot{X}}^T T_c$ and $\dot{Y} = L_{\dot{Y}}^T T_c$ according to the well known equations:

$$\begin{pmatrix} L_{\dot{X}}^T \\ L_{\dot{Y}}^T \end{pmatrix} = \begin{pmatrix} -\frac{1}{z} & 0 & \frac{X}{z} & XY & -1-X^2 & Y \\ 0 & -\frac{1}{z} & \frac{Y}{z} & 1+Y^2 & -XY & -X \end{pmatrix} \quad (12)$$

we obtain the expected expression:

$$L_{\underline{Q}}^T = M \begin{pmatrix} L_{X_1}^T - \alpha_1 L_{Y_1}^T \\ \vdots \\ L_{X_m}^T - \alpha_m L_{Y_m}^T \end{pmatrix} \quad (13)$$

and for a particular feature:

$$L_{Q_i}^T = \sum_{j=1}^{j=m} m_{i,j} \left(L_{X_j}^T - \alpha_j L_{Y_j}^T \right) \quad (14)$$

where $m_{i,j}$ denotes an element of M.

The use of (12), (5) and (7) in (14) yields the expressions given by (16) where we have : $\gamma_j = \sin\theta_j + \alpha_j \cos\theta_j$, $\varphi_i = \varphi_i(\underline{X}_c, \underline{\Theta}, \underline{\mathcal{M}}_i)$, $\psi_i = \psi_i(\underline{X}_c, \underline{\Theta}, \underline{\mathcal{M}}_i)$ with $\underline{X}_c = (X_c, Y_c)^T$, $\underline{\Theta} = (\theta_1, \ldots, \theta_m, \rho_1, \ldots, \rho_m)^T, \underline{\mathcal{M}}_i = (m_{i,1}, \ldots, m_{i,m})^T$ the expressions of which are complicated and not useful to be precised.

The relations given by (16) exhibit the terms $\sum_j m_{i,j}$, $\sum_j m_{i,j}\alpha_j$, $\sum_j m_{i,j}\beta_j\rho_j$ and $\sum_j m_{i,j}\gamma_j\rho_j$. We can verify, numerically or with a symbolic computation system, that these terms can be expressed, for any i and for any pairs (ρ_j, θ_j), by:

$$\begin{cases} \sum_j m_{i,j} & = 1, \underbrace{0, \ldots, 0}_{m-1\, times} \\ \sum_j m_{i,j}\alpha_j & = 0, -1, \underbrace{0, \ldots, 0}_{m-2\, times} \\ \sum_j m_{i,j}\beta_j\rho_j & = 0, 0, a_0, a_1, \ldots, a_h, b_1, \ldots, b_h \\ \sum_j m_{i,j}\gamma_j\rho_j & = 0, 0, 0, b_1, \ldots, hb_h, -a_1, \ldots, -ha_h \end{cases} \quad (15)$$

More, if we assume, for any $j = 1$ to m, $z_j = \hat{z}^*$, with $\hat{z}^*$ an approximation of the object depth at the desired location we obtain in (17) a simplified expression of the interaction matrix.

From equation (17), we can derive two important points:

- The first and second rows of $L_{\underline{Q}}^T\big|_{\hat{z}^*}$ show that the interaction matrix associated to the visual feature (X_c, Y_c) differs from that associated with the projection of a point (given by (12)) only by adding the terms φ_1, ψ_1, φ_2 and ψ_2, *i.e.* on terms related to the rotation around x and y axes.
- As expected, the visual features described by $\underline{P}$ do not vary during x or y translations of the robot. The shape of the contour is thus unchanged during such motions. Moreover, both translations are decoupled.

Concerning the behavior of the control law with regards to the number of harmonics, the first constraint for h is to achieve the positioning task. Thus, we have to ensure that $\widehat{L^T}_{|\underline{s}=\underline{s}^*}$ is full rank 6. It yields the necessary condition $2h+3 \geq 6$. So, the smaller value is theoretically $h = 2$. However, we will see in the next section that a higher value of h has to be chosen.

Once we have obtained the interaction matrix, we shall now examine how to extract the visual features in practice.

4. EXTRACTING THE VISUAL FEATURES

The problem is to obtain the values of the vector components given by:

$$\underline{s} = (X_c, Y_c, a_0, \cdots, a_h, b_1, \cdots, b_h)^T \quad (18)$$

In order to simplify the problem of image analysis and to focus only on the control law, we have used binary objects such as the object represented on Figure 1. In this case, computing (X_c, Y_c) is simple. We only discuss the way to proceed for the a_k's and the b_k's.

Theoretically, $2h + 1$ points extracted from the contour are sufficient. But, these points are necessary noised; moreover, we can assume that the number N of points available is such that $N \gg 2h + 1$. Thus, a least squares method has been used.

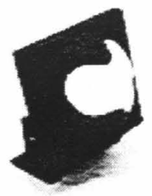

Fig. 1. The non planar object used.

We now return to the value of h that has to be selected. The visual features $\underline{s}$ appear in the algorithm at two places:

- in the control law: it must satisfy:

$$\underline{s}|_{k+1} = \underline{s}|_k + \Delta t L_{|\underline{s}}^T T_c \quad (19)$$

  where Δt is the image acquisition period.
- in the modelling of the image contour: the values of the visual features are computed to

$$\begin{aligned}
\widehat{L^T_{\underline{Q}_{i,1}}} &= \textstyle\sum_j m_{i,j}(-1/z_j)\\
\widehat{L^T_{\underline{Q}_{i,2}}} &= \textstyle\sum_j m_{i,j}\alpha_j/z_j\\
\widehat{L^T_{\underline{Q}_{i,3}}} &= X_c\textstyle\sum_j m_{i,j}/z_j - Y_c\sum_j \alpha_j m_{i,j}/z_j + \sum_j m_{i,j}\beta_j\rho_j\\
\widehat{L^T_{\underline{Q}_{i,4}}} &= X_cY_c\textstyle\sum_j m_{i,j} - (1+Y_c^2)\sum_j m_{i,j}\alpha_j + \varphi_i(\underline{X}_c, \underline{\Theta}, \underline{\mathcal{M}}_i)\\
\widehat{L^T_{\underline{Q}_{i,5}}} &= -(1+X_c^2)\textstyle\sum_j m_{i,j} + X_cY_c\sum_j m_{i,j}\alpha_j + \psi_i(\underline{X}_c, \underline{\Theta}, \underline{\mathcal{M}}_i)\\
\widehat{L^T_{\underline{Q}_{i,6}}} &= Y_c\textstyle\sum_j m_{i,j} + X_c\sum_j m_{i,j}\alpha_j + \sum_j m_{i,j}\gamma_j\rho_j
\end{aligned} \tag{16}$$

$$\left.\widehat{L^T_{\underline{Q}}}\right|_{\hat{z^*}} = \begin{pmatrix}
-1/\hat{z^*} & 0 & X_c/\hat{z^*} & X_cY_c+\varphi_1 & -1-X_c^2+\psi_1 & Y_c\\
0 & -1/\hat{z^*} & Y_c/\hat{z^*} & 1+Y_c^2+\varphi_2 & -X_cY_c+\psi_2 & -X_c\\
0 & 0 & a_0/\hat{z^*} & \varphi_3 & \psi_3 & 0\\
0 & 0 & a_1/\hat{z^*} & \varphi_4 & \psi_4 & b_1\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
0 & 0 & a_k/\hat{z^*} & \varphi_{k+3} & \psi_{k+3} & kb_k\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
0 & 0 & a_h/\hat{z^*} & \varphi_{h+3} & \psi_{h+3} & hb_h\\
0 & 0 & b_1/\hat{z^*} & \varphi_{h+4} & \psi_{h+4} & -a_1\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
0 & 0 & b_k/\hat{z^*} & \varphi_{k+h+3} & \psi_{k+h+3} & -ka_k\\
0 & 0 & b_h/\hat{z^*} & \varphi_{2h+3} & \psi_{2h+3} & -ha_h
\end{pmatrix} \tag{17}$$

minimize the distance between the contour and the modellized one.

These two points can be summarized by:

$$\underline{s}|_{k+1} = \underline{s}|_k + \Delta t L^T_{|\underline{s}} T_c + \delta\underline{s}|_{k+1} \tag{20}$$

where $\delta\underline{s}|_{k+1}$ is a term introduced to describe how the modelling between k and $k+1$ changes (if $h \longrightarrow \infty$, we have $\delta\underline{s}|_{k+1} \longrightarrow 0$).

Finally, h must be high enough to obtain the convergence of the control law so that (19) describes coarsely, but at each time k, the change of $\underline{s}$ when T_c is applied to the robot. Otherwise, local minima may occur. In such cases, the last and desired images are very similar but a positioning error exists. We could impose a very high value for h to ensure low enough $\delta\underline{s}|_{k+1}$, but it would lead to heavy computational time incompatible with a running at a video rate. Thus, since h has a finite value, the interaction matrix (17) is an approximation.

Typically, as will be shown in the experimental results, a value of h around 25 gives satisfactory results.

Once we have extracted the visual features, we can apply, according to (2), (3) and (4), the following control law:

$$T_c = -\lambda \left(\left. L^T_{\underline{s}} \right|_{z=\hat{z}^*, \underline{s}=\underline{s}^*} \right)^+ (\underline{s} - \underline{s}^*) \tag{21}$$

5. EXPERIMENTAL RESULTS

A complete description of the behavior of (21) is reported in (Collewet, 1999), only the most important results are detailed here.

The first experiment consisted of achieving a positioning task with regards to a planar object. The pose was charaterized by the Euler's angles, let us denotes γ, β, α these angles which respectively represent the x, y and z axis rotation (with the z axis corresponding to the optical axis). The initial pose was (in degrees): $\gamma = 31.21$, $\beta = 16.42$, $\alpha = -4.14$, the desired pose was $\gamma = 17.90$, $\beta = 0.06$, $\alpha = 0.31$ and the obtained one was $\gamma = 17.62$, $\beta = -0.01$, $\alpha = 0.28$. For this experiment h was fixed to 20 (this value led to Δt = 160 ms on a Pentium at 200 Mhz). Figure 2a shows the first image and the desired position of the object, Figure 2b the last image. Figures 2c and 2d show respectivelly the components of T_c and the normalized error (defined by $\frac{\|\underline{s}-\underline{s}^*\|_k}{\|\underline{s}-\underline{s}^*\|_0}$). These results show that the desired location was precisely reached without any particular problem.

The second experiment consisted of achieving a positioning task "in front of" a non planar object. The maximum difference of depth between contour points was nearly 4 cm (see Figure 1). In this case, satisfactory results were obtained with h = 25 (Δt = 240 ms) but not for h = 20 (h = 25 was also suitable for the planar object). The initial pose was $\gamma = 17.67$, $\beta = 19.29$, $\alpha = 16.86$ and the final pose was $\gamma = 0.36$, $\beta = 0.12$, $\alpha = -0.33$ (instead of $\gamma = \beta = \alpha = 0$). Results are depicted on Figure 3. Again, although we supposed in (21) that the object was planar and parallel to the image plane ($\forall j,\ z_j = \hat{z}^*$), the desired location was reached. Our approach is thus robust to such approximations.

We next introduced errors in the camera model and in the transformation matrix between the robot and the camera. In (Collewet, 1999), we show that these errors have low effect on the behavior of the control scheme. In the same way, a wide range of $\hat{z}^*$ (from 30 cm to 250 cm) led to the desired location. More, as we can see on figure 4, a partial occlusion was not critical at the beginning of the motion. However we cannot give any results when the occlusion appears, for example, in the middle of the motion. We did not succeed in perfecting such an experiment.

6. SUMMARY AND CONCLUSIONS

In this paper, we focused in the way to achieve positioning tasks with respect to objects of complex shape. We have analytically computed an approximation of the interaction matrix associated with a polar description of the object contour. The experimental results validated the proposed approach. They showed that convergence can be obtained under very low hypotheses (but still in the usual cases of application of 2D visual servoing, *i.e.* when $\underline{s} - \underline{s}^*$ is not to large). Only current and desired images are necessary, no precise knowledge about the object shape and range is required. More, experimental results have shown the robustness of the control law with regards to approximations of the depths, certain partial occlusion and a coarsely calibrated system.

Besides, if we compare our approach to an approach based on interest points (see for example (Schmid *et al.*, 1998) for the extraction of such points), since we can use many contour points, noise has low effect on our control law. On the other hand, a further problem might occur, *i.e.* the matching of points between images. This problem is complicated, especially between very different images such as the first and the desired image. This problem does not exist with our contour-based approach.

However, we can regret heavy computational cost of our method when h is high and the consequences on the dynamic performances of the robot (typically, we recall that $\Delta t = 240$ ms for $h = 25$ and a binary object). In the future, it would be interesting to combine the computation of the visual features obtained from the contour and its extraction. A suitable approach could consist in using parametric active contours like those described in (Drummond and Cipolla, 1999; Colombo and Allotta, 1999). Our approach based on a polar description should be integrated into this work without any particular difficulty.

7. REFERENCES

Bien, Z., W. Jang and J. Park (1993). Characterisation and use of feature-jacobian matrix for visual servoing. In: *Visual Servoing* (K. Hashimoto, Ed.). Vol. 7. pp. 317–363. World Scientific. Singapore.

Collewet, C. (1999). Contributions à l'élargissement du champ applicatif des asservissements visuels 2D. PhD thesis. Rennes I University. France.

Colombo, C. and B. Allotta (1999). Image-based robot task planning and control using a compact visual representation. *IEEE Trans. on Systems, Man, and Cybernetics — Part A: Systems and Humans* **29**(1), 92–100.

Deguchi, K. and T. Noguchi (1996). Visual servoing using eigenspace method and dynamic calculation of interaction matrices. In: *Int. Conf. on Pattern Recognition, ICPR'96.* pp. 302–306.

Drummond, T.W. and R. Cipolla (1999). Visual tracking and control using lie algebras. In: *IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR'99.* Vol. II. Fort Collins, Colorado. pp. 652–657. USA.

Espiau, B., F. Chaumette and P. Rives (1992). A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation* **8**(3), 313–326.

Feddema, J. T., C. S. G. Lee and O. R. Mitchell (1989). Automatic selection of image features for visual servoing of a robot manipulator. In: *IEEE Int. Conf. on Robotics and Automation, ICRA'89.* Washington. pp. 832–837. USA.

Hutchinson, S., G. D. Hager and P. I. Corke (1996). A tutorial on visual servo control. *IEEE Trans. on Robotics and Automation* **12**(5), 651–670.

Nayar, S. K., S. A. Nene and H. Murase (1996). Subspace methods for robot vision. *IEEE Trans. on Robotics and Automation* **12**(5), 750–758.

Schmid, C., R. Mohr and C. Bauckhage (1998). Comparing and evaluating interest points. In: *6th Int. Conf. on Computer Vision, ICCV'98.* Bombay, India.

Wells, G., C. Venaille and C. Torras (1996). Promising research vision-based robot positioning using neural networks. *Image and Vision Computing* **14**, 715–732.

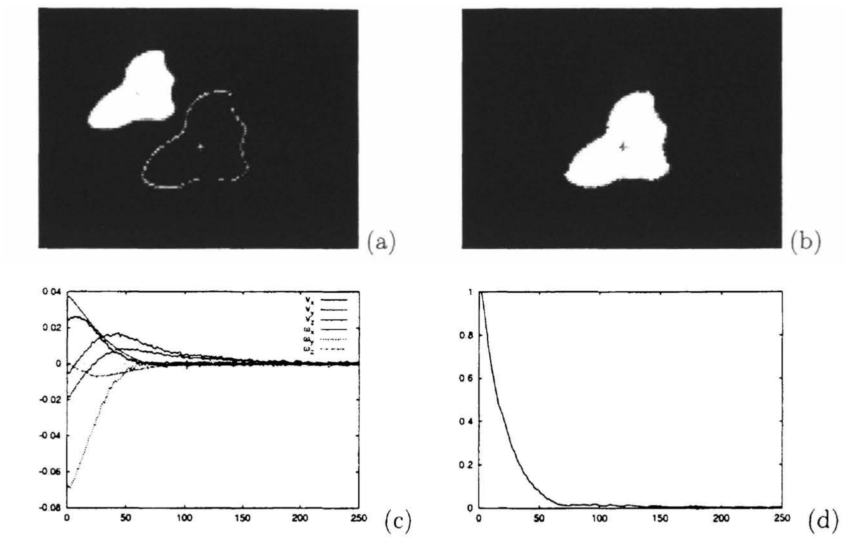

Fig. 2. Positioning task with regards to a planar object: (a) First and desired images, (b) Last image, (c) Components of T_c (m/s and rad/s), (d) Normalized error.

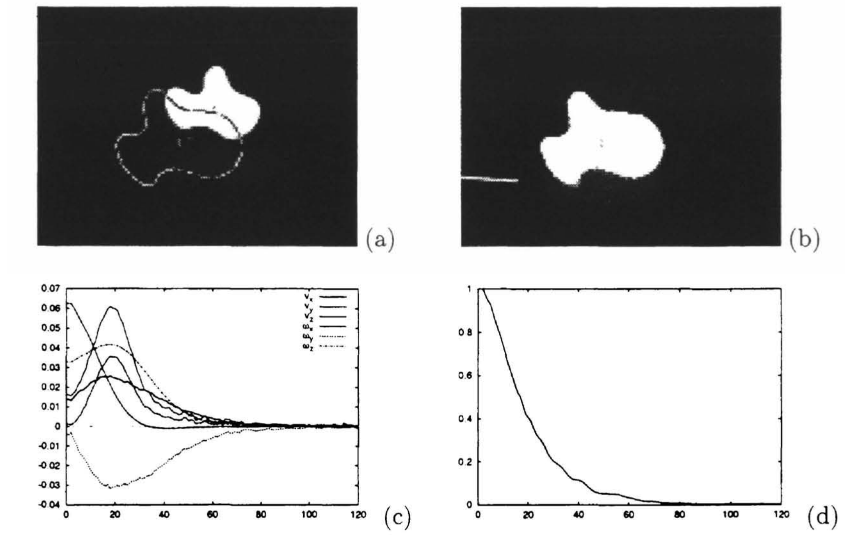

Fig. 3. Positioning task with regards to a non planar object.

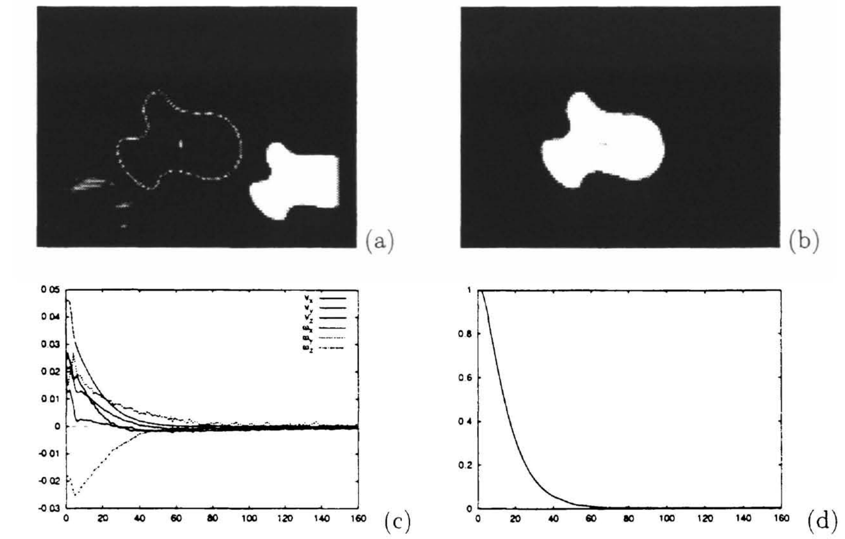

Fig. 4. Occlusion test on a non planar object.