ROBOT GUIDANCE BY ESTIMATING THE FORCE-IMAGE INTERACTION MATRIX

G. J. García, J. Pomares, F. Torres

Physics, Systems Engineering and Signal Theory Department
University of Alicante
PO Box 99, 03080, Alicante. Spain.
{gjgg, jpomares, Fernando.Torres}@ua.es

Abstract: This paper describes an uncalibrated visual-force control system which does not require any kinematic calibration to develop the task. An important aspect of these kinds of control systems is the necessity to maintain the coherence between the control actions obtained from each sensorial system. To do so, the paper proposes to modify the image trajectory from the information obtained from the force sensor by using the concept of force-image interaction matrix. This matrix relates changes in the image space with changes in the interaction forces. In order to estimate the value of this matrix this paper suggests the use of a Gauss-Newton method.

Keywords: visual servoing, force control, sensor fusion, estimation algorithms, robot vision

1. INTRODUCTION

Most actual visual servoing schemes (Hutchinson *et al*., 1996) require a previous calibration in order to determine the camera intrinsic parameters of the camera, kinematics or dynamics of the robot, etc. Nowadays, visual servoing systems are employed in a great number of applications and also in very different environments. With the objective of increasing the flexibility of these systems it is possible to emphasize works guided to fit with robustness in the presence of uncertainty in certain calibration parameters like the intrinsic ones (Malis, 2004). The necessity of performing a previous calibration is even more obvious in visual-force control systems. In this type of control systems the robot simultaneously carries out a visual control task and an interaction control with the environment. In visual-force control tasks previously unknown parameters such as frictions, relationships between sensor coordinate frames, properties of the contact surface, etc. appear.

In this article a visual-force control system that does not require a previous calibration between vision and force sensor systems is presented. In order to combine the sensorial information, a modification of the trajectory in the image tracked by the visual servoing system is performed based on the information obtained from the force sensor. To do this, in this article, the force-image interaction matrix concept has been defined. This matrix determines the relationship between the variations in the image space and the variations in the interaction forces. This paper presents a method which is able to obtain an online estimation of the force-image interaction matrix without the necessity of making use of previous knowledge of the calibrated parameters. The method makes an estimation based on the information obtained from both sensors. The system employed in the estimation is based on the recursive Gauss-Newton method and uses nonlinear least-squares optimization. This method is based on previous works like (Hosoda and Asada, 1994; Jagersand *et al*., 1997; Piepmeier and Lipkin, 2003) which develops a dynamic Gauss-Newton method of a visual servo control to track moving objects. These methods are undertaken to obtain a robust estimation of the image Jacobian, which relates visual features with the variation in the robot joint coordinates.

This paper is organized as follows: The force-image interaction matrix is defined in Section II. Section III describes the visual-force control system. In Section IV, the method to estimate the force-image interaction matrix is presented. In Section V, experimental results, using an eye-in-hand camera system, confirm the validity of the proposed algorithms. The final section presents the main conclusions arrived at.

2. IMAGE – FORCE INTERACTION MATRIX

In this section the meaning of the force-image interaction matrix, $\mathbf{L}_{FI}$, is described. To do so, considering $\mathbf{F}$ the interaction forces obtained with respect to the robot end-effector and $\mathbf{r}$ the object location, the interaction matrix for the interaction forces, $\mathbf{L}_F$, is defined in this way:

$$\mathbf{L}_F = \frac{\partial \mathbf{F}}{\partial \mathbf{r}} \rightarrow \mathbf{L}_F^+ = \left(\mathbf{L}_F^T \mathbf{L}_F\right)^{-1} \mathbf{L}_F^T \frac{\partial \mathbf{r}}{\partial \mathbf{F}} \qquad (1)$$

Furthermore, considering the joint coordinates of the robot as $\mathbf{q}$, the Jacobian of the robot, $\mathbf{J}$, follows this relation:

$$\boldsymbol{v}_e = \dot{\mathbf{r}} = \mathbf{J} \cdot \dot{\mathbf{q}} \rightarrow \dot{\mathbf{q}} = \mathbf{J}^+ \cdot \dot{\mathbf{r}} \qquad (2)$$

where $\boldsymbol{v}_e = \left[\dot{\mathbf{p}}_e \quad \boldsymbol{\omega}_e\right]$, $\dot{\mathbf{p}}_e$ is the 3x1 vector of end-effector linear velocity, and $\boldsymbol{\omega}_e$ the 3x1 vector of end-effector angular velocity.

With $\mathbf{J}_I$ is represented the image Jacobian (Hutchinson *et al*., 1996) which relates the variations in the image with the variation in the joint space, i.e., $\dot{\mathbf{s}} = \mathbf{J}_I \cdot \dot{\mathbf{q}}$, where $\mathbf{s}=[\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_M]^T$ is the set of features extracted from the image. Through this last relationship and by applying (2) is obtained:

$$\begin{aligned} \dot{\mathbf{s}} &= \mathbf{J}_I \cdot \mathbf{J}^+ \cdot \frac{\partial \mathbf{r}}{\partial t} = \mathbf{J}_I \cdot \mathbf{J}^+ \cdot \frac{\partial \mathbf{r}}{\partial \mathbf{F}} \cdot \frac{\partial \mathbf{F}}{\partial t} \\ \dot{\mathbf{s}} &= \mathbf{J}_I \cdot \mathbf{J}^+ \cdot \mathbf{L}_F^+ \cdot \dot{\mathbf{F}} \rightarrow \dot{\mathbf{s}} = \mathbf{L}_{FI} \cdot \dot{\mathbf{F}} \end{aligned} \qquad (3)$$

where $\mathbf{L}_{FI}$ is the interaction matrix to be estimated using exponentially weighted least-squares.

3. FORCE-IMAGE CONTROL

Now, we consider the task of tracking a surface using visual and force information. The visual loop carries out the tracking of the desired trajectory in the image space. To do so, an image-based control scheme to regulate to 0 the following vision-based task function is used (Mezouar and Chaumette, 2005):

$$\boldsymbol{e} = \mathbf{J} \cdot \hat{\mathbf{J}}_I^+ \cdot \left(\mathbf{s} - \mathbf{s}_d(t)\right) \qquad (4)$$

where $\mathbf{s}$ are the features extracted from the image, $\hat{\mathbf{J}}_I^+$ is an estimation of the pseudoinverse of the previous defined image Jacobian and $\mathbf{s}_d(t)$ is the desired trajectory in the image.

Now, we consider $\gamma(t)$ the desired trajectory to be tracked in the 3-D space. By sampling $\gamma(t)$ (considering that the time between two consecutive samples is constant, so that $\Delta\, {}^{k+1}t = {}^{k+1}t - {}^{k}t = T$ where T is the video rate), a sequence of N discrete values is obtained, each of which represents N intermediate positions of the camera $\left\{{}^{k}\boldsymbol{\gamma}/k \in 1...N\right\}$. From this sequence, the discrete trajectory of the object in the image $S = \left\{{}^{k}\mathbf{s}/k \in 1...N\right\}$ can be obtained (see (Pomares and Torres, 2005)), where ${}^{k}\mathbf{s}$ is the set of M points or features observed by the camera at instant k, ${}^{k}\mathbf{s} = \left\{{}^{k}\boldsymbol{f}_i/i \in 1...M\right\}$.

From $S = \left\{{}^{k}\mathbf{s}/k \in 1...N\right\}$ and a set of parameter values $\Gamma = \left\{{}^{k}t/k \in 1...N\right\}$ the desired trajectory $\mathbf{s}_d(t)$ to be tracked in the image is obtained using a natural cubic B-spline such that $\mathbf{s}_d(t_k) = {}^{k}\mathbf{s}$. To carry out the tracking of the trajectory using a classical image based visual servoing system, the following velocity can be applied to the robot:

$$\boldsymbol{v}_V = -\lambda_V \cdot \boldsymbol{e} + \mathbf{J} \cdot \hat{\mathbf{J}}_I^+ \cdot \frac{\partial \mathbf{s}_d(t)}{\partial t} \qquad (5)$$

where $\lambda_V > 0$ is the gain of the proportional controller.

As it has been described in our previous works (Pomares and Torres, 2005), in order to guarantee the coherence between the visual and force information it is necessary to modify the image trajectory through the interaction forces. Therefore, in an application in which it is necessary to maintain a constant force with the workspace, the image trajectory must be modified depending on the interaction forces. To do so, using the matrix $\mathbf{L}_{FI}$, the new desired features used by the controller (5) during the contact will be:

$$\mathbf{s}_d'(t) = \mathbf{s}_d(t) + \mathbf{L}_{FI} \cdot \left(\mathbf{F} - \mathbf{F}_d\right) \qquad (6)$$

4. FORCE-IMAGE INTERACTION MATRIX ESTIMATION

In this section the method used for the estimation of the interaction matrix $\mathbf{L}_{FI}$ will be described. The algorithm proposed to the estimation is based on the Gauss-Newton method which has been previously applied to the estimation of the interaction matrix in image based visual servoing systems (Hosoda and Asada, 1994; Jagersand *et al*., 1997; Piepmeier and Lipkin, 2003).

Considering a MIMO system in which the inputs are variations in the interaction forces and the outputs are variations in the visual features, the measurement equation can be represented in the following way:

$$\Delta \mathbf{s} = \mathbf{L}_{FI} \cdot \Delta \mathbf{F} + \mathbf{e} \qquad (7)$$

where:

- $\Delta \mathbf{s} = \left[\Delta s_1 \;\; \ldots \;\; \Delta s_m\right]^T$ is the output of the system, i.e., the variation in the visual features.

- $\Delta\mathbf{F} = \left[\Delta F_1 \;\; \ldots \;\; \Delta F_n\right]^{T}$ is the input of the system, i.e., the variation in the interaction forces.
- $\mathbf{e}$ is a measure error vector (white noise).
- $\mathbf{L}_{FI}$ is the force-image interaction matrix previously defined whose estimated value will be represented by $\hat{\mathbf{L}}_{FI}$ (for the sake of clarity in notation, from now on $\mathbf{L}_{FI}$ is represented as $\mathbf{L}$):

$$\mathbf{L} = \begin{pmatrix} L_{11} & \ldots & L_{1n} \\ \ldots & \ldots & \ldots \\ L_{m1} & \ldots & L_{mn} \end{pmatrix}$$
$$\hat{\mathbf{L}}_j = \left[\hat{L}_{j1} \;\; \ldots \;\; \hat{L}_{jn}\right]^{T}, j = 1 \ldots m \tag{8}$$

Therefore, the value for each of the m outputs or visual features will be:

$$\Delta s_j = \hat{\mathbf{L}}_j^{T} \cdot \Delta\mathbf{F} + e_j \tag{9}$$

From Equation (9) and considering k measures, the following equation for the exponentially weighted error is obtained:

$$\begin{pmatrix} \lambda^{(k-1)/2} e_j(1) \\ \lambda^{(k-2)/2} e_j(2) \\ \vdots \\ e_j(k) \end{pmatrix} = \begin{pmatrix} \lambda^{(k-1)/2} \Delta s_j(1) \\ \lambda^{(k-2)/2} \Delta s_j(2) \\ \vdots \\ \Delta s_j(k) \end{pmatrix} - \begin{pmatrix} \lambda^{(k-1)/2} (\Delta\mathbf{F}(1))^{T} \\ \lambda^{(k-2)/2} (\Delta\mathbf{F}(2))^{T} \\ \vdots \\ (\Delta\mathbf{F}(k))^{T} \end{pmatrix} \cdot \hat{\mathbf{L}}_j \tag{10}$$

where the parameter $\lambda \in (0,1]$ indicates the weight of the new measures. The higher the value of λ are, the more weight is assigned to the new measures. Equation (10) can be represented in the following way too:

$$\bar{\mathbf{e}}_{jk} = \bar{\mathbf{S}}_{jk} - \bar{\mathbf{F}}_k \cdot \hat{\mathbf{L}}_j \tag{11}$$

With the objective to estimate the force-image interaction matrix the following objective function to minimize is defined:

$$E_k = \sum_{j=1}^{m} \bar{\mathbf{e}}_{jk}^{T} \bar{\mathbf{e}}_{jk} = \sum_{j=1}^{m} \left(\bar{\mathbf{S}}_{jk} - \bar{\mathbf{F}}_k \cdot \hat{\mathbf{L}}_j\right)^{T} \left(\bar{\mathbf{S}}_{jk} - \bar{\mathbf{F}}_k \cdot \hat{\mathbf{L}}_j\right) \tag{12}$$

The estimation of the force-image interaction matrix can be represented in the following way:

$$\hat{\mathbf{L}} = \left[\hat{\mathbf{L}}_1^{T} \;\; \cdots \;\; \hat{\mathbf{L}}_m^{T}\right]^{T} \tag{13}$$

The value of $\hat{\mathbf{L}}_j(k+1)$ can be deduced from the measure k+1 and the value of $\hat{\mathbf{L}}_j(k)$. For k+1 measures, the exponentially weighted error will be the following:

$$\begin{pmatrix} \sqrt{\lambda}\,\bar{\mathbf{e}}_{jk} \\ e_j(k+1) \end{pmatrix} = \begin{pmatrix} \sqrt{\lambda}\,\bar{\mathbf{S}}_{jk} \\ \Delta s_j(k+1) \end{pmatrix} - \begin{pmatrix} \sqrt{\lambda}\,\bar{\mathbf{F}}_k \\ (\Delta\mathbf{F}(k+1))^{T} \end{pmatrix} \cdot \hat{\mathbf{L}}_j \tag{14}$$

Furthermore, using the dynamic Gauss-Newton method [6], the following condition must be fulfilled if it is desired in the k+1 iteration to completely reduce the error represented in the Equation (12):

$$\hat{\mathbf{L}}_j(k+1) = \left(\bar{\mathbf{F}}_{k+1}^{T} \bar{\mathbf{F}}_{k+1}\right)^{-1} \bar{\mathbf{F}}_{k+1}^{T} \bar{\mathbf{S}}_{jk+1} \tag{15}$$

Suppose now that:

$$\mathbf{P}(k) = \left(\bar{\mathbf{F}}_k^{T} \bar{\mathbf{F}}_k\right)^{-1} \tag{16}$$

Therefore:

$$\mathbf{P}(k+1) = \left(\bar{\mathbf{F}}_{k+1}^{T} \bar{\mathbf{F}}_{k+1}\right)^{-1} = \left[\left(\sqrt{\lambda}\,\bar{\mathbf{F}}_k^{T} \;\; \Delta\mathbf{F}(k+1)\right) \begin{pmatrix} \sqrt{\lambda}\,\bar{\mathbf{F}}_k \\ (\Delta\mathbf{F}(k+1))^{T} \end{pmatrix}\right]^{-1} \tag{17}$$

By multiplying the terms in Equation (17) is obtained:

$$\mathbf{P}(k+1) = \left[\lambda \bar{\mathbf{F}}_k^{T} \bar{\mathbf{F}}_k + \Delta\mathbf{F}(k+1)(\Delta\mathbf{F}(k+1))^{T}\right]^{-1} = \frac{1}{\lambda}\left[\mathbf{P}^{-1}(k) + \Delta\mathbf{F}(k+1)\lambda^{-1}(\Delta\mathbf{F}(k+1))^{T}\right]^{-1} \tag{18}$$

Finally, by developing the above expression it follows that:

$$\mathbf{P}(k+1) = \frac{1}{\lambda}[\mathbf{P}(k) - \mathbf{P}(k)\Delta\mathbf{F}(k+1) \cdot \left[\lambda + (\Delta\mathbf{F}(k+1))^{T} \mathbf{P}(k) \Delta\mathbf{F}(k+1)\right]^{-1} \cdot (\Delta\mathbf{F}(k+1))^{T} \mathbf{P}(k)] \tag{19}$$

With the objective of obtaining a more compact representation of the previous expression $\mathbf{K}(k+1)$ is defined as:

$$
\mathbf{K}(k+1) = \mathbf{P}(k)\,\Delta\mathbf{F}(k+1)\cdot
\cdot\left[\lambda + \Delta\mathbf{F}(k+1)^{T}\mathbf{P}(k)\,\Delta\mathbf{F}(k+1)\right]^{-1} \tag{20}
$$

Therefore, applying (20) in (19) it can be concluded that:

$$
\mathbf{P}(k+1) =
\frac{1}{\lambda}\left[\mathbf{P}(k) - \mathbf{K}(k+1)\,\Delta\mathbf{F}(k+1)^{T}\mathbf{P}(k)\right] \tag{21}
$$

Moreover, through the Equation (15) it can be deduced that:

$$
\hat{\mathbf{L}}_j(k+1) =
= \mathbf{P}(k+1)\left[\sqrt{\lambda}\bar{\mathbf{F}}_k^{T} \quad \Delta\mathbf{F}(k+1)\right]\left[\begin{matrix}\sqrt{\lambda}\bar{\mathbf{S}}_{jk} \\ \Delta s_j(k+1)\end{matrix}\right] =
\mathbf{P}(k+1)\left[\lambda\bar{\mathbf{F}}_k^{T}\bar{\mathbf{S}}_{jk} + \Delta\mathbf{F}(k+1)\,\Delta s_j(k+1)\right] \tag{22}
$$

Applying (21) in (22):

$$
\hat{\mathbf{L}}_j(k+1) = \hat{\mathbf{L}}_j(k) + \mathbf{K}(k+1)\cdot
\cdot\left[\Delta s_j(k+1) - \Delta\mathbf{F}(k+1)^{T}\hat{\mathbf{L}}_j(k)\right] \tag{23}
$$

From (23) it is obtained:

$$
\hat{\mathbf{L}}_j^{T}(k+1) = \hat{\mathbf{L}}_j^{T}(k) +
+\left[\Delta s_j(k+1) - \hat{\mathbf{L}}_j^{T}(k)\,\Delta\mathbf{F}(k+1)\right]\mathbf{K}^{T}(k+1) \tag{24}
$$

Finally, by grouping the estimations of the m rows of the interaction matrix it is obtained:

$$
\hat{\mathbf{L}}^{T}(k+1) = \hat{\mathbf{L}}^{T}(k) +
+\left[\Delta \mathbf{s}(k+1) - \hat{\mathbf{L}}^{T}(k)\,\Delta\mathbf{F}(k+1)\right]\mathbf{K}^{T}(k+1) \tag{25}
$$

where the value of the matrix **K** is obtained from the expression (20). Therefore, using Equation (25) recursively, an estimation of the force-image interaction matrix is obtained.

Summarizing the different steps required to determine the value of the force-image interaction matrix, we can obtain the following algorithm:

$\Delta\mathbf{s}: \Re^{n} \rightarrow \Re^{m}$; $\Delta F_0,\ \Delta F_1 \in \Re^{n}$; $\hat{\mathbf{L}}(0) \in \Re^{mxn}$;
$\hat{\mathbf{P}}(0) \in \Re^{nxn}$; $\lambda \in (0,1)$

For k=0, 1, ... do

$\Delta\mathbf{s}(k+1) = \mathbf{s}_{k+1} - \mathbf{s}_k$

$\Delta\mathbf{F}(k+1) = \mathbf{F}_{k+1} - \mathbf{F}_k$

$\hat{\mathbf{L}}(k+1) = \hat{\mathbf{L}}(k) +$
$+\left[\Delta\mathbf{s}(k+1) - \hat{\mathbf{L}}(k)\,\Delta\mathbf{F}(k+1)\right]\mathbf{K}^{T}(k+1)$

$\mathbf{K}(k+1) = \mathbf{P}(k)\,\Delta\mathbf{F}(k+1)\cdot$
$\cdot\left[\lambda + \Delta\mathbf{F}(k+1)^{T}\mathbf{P}(k)\,\Delta\mathbf{F}(k+1)\right]^{-1}$

$\mathbf{P}(k+1) =$
$= \frac{1}{\lambda}\left[\mathbf{P}(k) - \mathbf{K}(k+1)\,\Delta\mathbf{F}(k+1)^{T}\mathbf{P}(k)\right]$

end

5. RESULTS

5.1 Simulation result.

With the objective to apply the method proposed, a simulated experiment, in which the visual features are varying according to the increase of the interaction forces in z, will be firstly described. This experiment describes a simulation of a descent in depth in which the interaction forces in the z axis increase linearly (1 N. in each iteration) and the features follow the evolution shown in Fig. 1 (a pixel in (x,y) at each iteration).

The initial values considered for the force-image interaction matrix $\hat{\mathbf{L}}(0)$ and $\hat{\mathbf{P}}(0)$ are respectively:

$$
\hat{\mathbf{L}}(0) = \begin{bmatrix} 0 & 0 & 0{,}001 \\ 0 & 0 & 0{,}001 \\ 0 & 0 & 0{,}001 \\ 0 & 0 & 0{,}001 \\ 0 & 0 & 0{,}001 \\ 0 & 0 & 0{,}001 \\ 0 & 0 & 0{,}001 \\ 0 & 0 & 0{,}001 \end{bmatrix} \quad \hat{\mathbf{P}}(0) = \begin{bmatrix} 9 & 0 & 0 \\ 0 & 9 & 0 \\ 0 & 0 & 9 \end{bmatrix} \tag{26}
$$

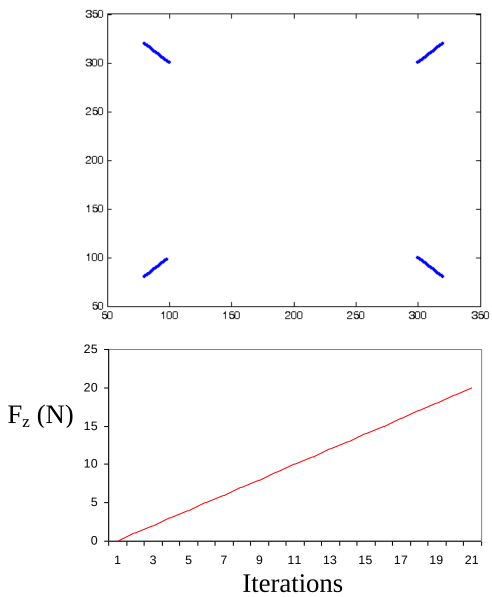

Fig. 1. Evolution of the image features and the interaction forces during the experiment.

In 3 iterations the value obtained is already very near to the adequate for the interaction matrix (λ = 0.9):

$$\hat{\mathbf{L}}\ 3\ = \begin{bmatrix} 0 & 0 & -0.9709 \\ 0 & 0 & -0.9709 \\ 0 & 0 & 0.9710 \\ 0 & 0 & -0.9709 \\ 0 & 0 & 0.9710 \\ 0 & 0 & 0.9710 \\ 0 & 0 & -0.9709 \\ 0 & 0 & 0.9710 \end{bmatrix} \tag{27}$$

The real interaction matrix employed in the simulations is:

$$\mathbf{L} = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & -1 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \\ 0 & 0 & 1 \end{bmatrix} \tag{28}$$

5.2 Experimental result.

In this experiment, the experimental assembly shown in Fig. 2 has been carried out. The system architecture is composed of an eye-in-hand PHOTONFOCUS MV-D752-160-CL-8 camera at the end-effector of a 7 d.o.f. Mitsubishi PA-10 robot also equipped with a force sensor. The camera is able to acquire and to process up to 100 frames/second using an image resolution of 320x240. In this paper we are not interested in image processing issues. Therefore, the image trajectory is generated by using four grey marks whose centres of gravity will be the extracted features.

During the experiments the robot collides with a table which presents a certain level of compressibility. This way, depending on the force with which the robot interacts with its environment, the system is able to clearly observe a modification of the features in the image. Through this information, the value of the force-image interaction matrix is determined as it was described in Section 4. This matrix is employed to modify the desired trajectory in the image so that a constant interaction force between the robot and the surface can be maintained. As it has been described in previous works (Pomares and Torres, 2005) the modification of the visual information through the information of interaction with the environment guarantees the coherence between both sensorial systems. This way, the production of contradictory control actions between both sensorial systems is avoided. In these experiments, two simple examples are illustrated in which it is necessary to maintain constant force between the robot and the contact surface. However, in these experiments the correct behaviour of the proposed system is clearly observed.

Experiment 1. In this experiment the robot is carrying out the tracking of a trajectory using a visual servoing system (see Equation (5)). However during the trajectory the robot collides with the table illustrated in Fig. 2. Using the algorithm described in Section 4 the system determines the value of the force-image interaction matrix obtaining the matrix represented in Equation (29) (λ = 0.95). Finally, considering 20 N. as the desired contact force in z direction, the evolution of the interaction forces is represented in Fig. 3. This figure shows the correct behaviour of the visual-force control system.

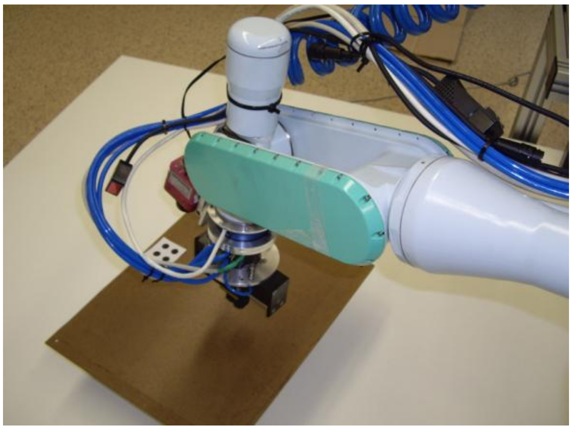

Fig. 2. Experimental setup.

In this experiment it is possible to observe that the system maintains constant force with the interaction surface by using only visual information and the estimated force-image interaction matrix. From the interaction forces a variation of the visual features are obtained using the force-image interaction matrix. Therefore, the visual controller proposed in Section 3 can be used for controlling the robot interaction and also for controlling the tracking of the trajectory only using visual information. This aspect guarantees that the visual and force information can be jointly used to control the robot during the task.

$$\hat{\mathbf{L}} = \begin{bmatrix} 0.6079 & -0.6347 & -0.0469 \\ -0.6079 & 0.6347 & 0.0469 \\ -0.6079 & 0.6347 & 0.0469 \\ -0.6079 & 0.6347 & 0.0469 \\ -0.6079 & 0.6347 & 0.0469 \\ 0.6079 & -0.6347 & -0.0469 \\ 0.6079 & -0.6347 & -0.0469 \\ 0.6079 & -0.6347 & -0.0469 \end{bmatrix} \tag{29}$$

Experiment 2. In this experiment the robot is also tracking an image trajectory using the previous mentioned visual controller. However, in this case during the interaction the robot must maintain a constant force of 40 N. in z direction against the contact surface.

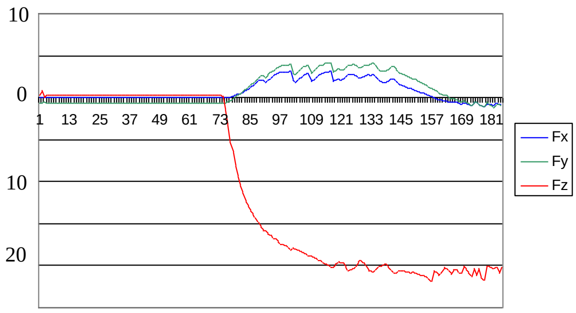

Fig. 3. Evolution of the interaction forces during the experiment. Experiment 1.

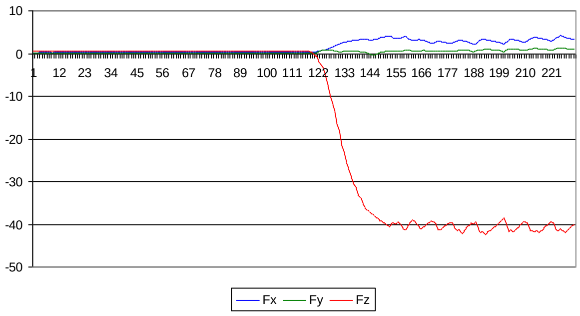

Fig. 4. Evolution of the interaction forces during the experiment. Experiment 2.

The same interaction surface is used so that the force-image interaction matrix indicated in Equation (29) is also considered. With these considerations the interaction forces represented in Fig. 4 are obtained when the robot collides with the contact surface. In this figure it is possible to observe that the system is able to maintain the constant force of 40 N. between the robot and the surface in z direction. Therefore the visual information is employed not only to control the tracking of the trajectory but also to control the robot interaction with the workspace.

6. CONCLUSIONS

In this article, a new uncalibrated scheme for vision-force robotic guidance has been presented. This method does not require previous knowledge of the robot's kinematic model, camera calibration or sensorial calibration between the camera and the force sensor. In applications in which different sensorial information is combined it is necessary to avoid situations in which contradictory control actions are obtained from each sensor. To do so, in this paper, a new method to modify image trajectories from the information of the force sensor is proposed. This method is based on the estimation of the force-image interaction matrix. This matrix relates changes in the image space with changes in the interaction forces. In order to estimate the value of this matrix a method based on Gauss-Newton algorithm is applied.

ACKNOWLEDGEMENTS

This work was funded by the Spanish MEC project "Diseño, implementación y experimentación de escenarios de manipulación inteligentes para aplicaciones de ensamblado y desensamblado automático".

REFERENCES

[1] Hosoda, K. and Asada, M. (1994). Versatile visual servoing without knowledge of true Jacobian, In IEEE/RSJ/GI International Conference on Intelligent Robots and Systems, September, Munich, Germany, pp. 186-193.

[2] Hutchinson, S., Hager, G., Corke, P., (1996). A Tutorial on Visual Servo Control. *IEEE Trans. on Robotics and Automation*, **12(5)**, 651-670.

[3] Jagersand, M., Fuentes, O., and Nelson, R. (1997). Experimental evaluation of uncalibrated visual servoing for preceision manipulation. In Proceedings of International Conference on Robotics and Automation, April, Albuquerque; NM, pp. 2874-2880.

[4] E.Malis (2004). Visual servoing invariant to changes in camera intrinsic parameters. *IEEE Transaction on Robotics and Automation*, **20(1)**, 72-81.

[5] Y. Mezouar and F. Chaumette (2002). Path Planning For Robust Image-based Control. *IEEE Trans. on Robot. Automat*., **18(4)**, 534-549.

[6] Piepmeier, J. A., Lipkin, H. (2003). Uncalibrated eye-in-hand visual servoing. *International Journal of Robotics Research*. **22(10-11)**, 805-819.

[7] J. Pomares, F. Torres. (2005). Movement-flow based visual servoing and force control fusion for manipulation tasks in unstructured environments. *IEEE Transactions on Systems, Man, and Cybernetics—Part C*, **35(1)**, 4 – 15.