# Robot Control Using Disparate Multiple Sensors

**Ezio Malis**

Institut National de la Recherche
en Informatique et Automatique
2004, route des Lucioles, 06902
Sophia-Antipolis, France

**Guillaume Morel**

Electricité de France Research and Development
Simulation et Optimisation des Chantiers de Maintenance
Chatou, France

**François Chaumette**

Institut de Recherche en Informatique et Systèmes
Aléatoires/Institut National de la Recherche en
Informatique et Automatique
Campus de Beaulieu, 35042
Rennes, France

## Abstract

*A simple and efficient control algorithm that combines several sensors in order to realize the positioning task of a robot end effector is presented in this paper. The multiple-sensor control was designed as a part of the task function approach. A particular choice of the task function allows the researcher to simplify the design of the control law and the stability analysis. This global controller is based on the weighted sum of individual task functions. The approach was applied to the control with vision and force sensors. In spite of its simplicity, this approach provides satisfactory experimental results. Improvements in the positioning of cumbersome objects were obtained using cameras observing different parts of a scene. Moreover, peg-in-hole insertion experiments involving large initial errors were performed using a seven-axis robot manipulator without any computation of the peg trajectory by combining vision and force sensors.*

KEY WORDS—multisensor, force, vision, control, robotics

## 1. Introduction

Most applications of advanced robotics need robot manipulators with the ability to work in environments with unknown location and geometry. Thus, external sensory information has to be integrated in the manipulator control. Over the past decades, force/torque sensors, proximity sensors, and video cameras have been used to control manipulators in unknown environments. Early in robot control development, force-sensing capabilities were considered to be a crucial issue, since the robot often interacts with its environment. Much research has been conducted to study stability properties and bandwidth limitation and to emphasize the role of dynamics in force control (Eppinger and Seering 1989; Khatib 1987; Raibert and Craig 1981; Lawrence 1988). Vision-based control has recently received a growing interest as the computational power of commercially available computers has become compatible with real-time visual feedback (Espiau, Chaumette, and Rives 1992; Hutchinson, Hager, and Corke 1996; Wilson, Hulls, and Bell 1996). For both vision and force control, initial drawbacks have been overcome and a number of techniques are now available. They should be selected depending on the nature of the task, the robot and sensor design, and the low-level controller hardware. At a more general level, the task function approach was proposed in Samson, Le Borgne, and Espiau (1991) as a general framework for sensor-based control of robots.

Combining several sensory data is also an important issue that has been studied considering two fundamentally different approaches. In the first approach, the different sensors complementarily measure the same physical phenomena. Thus, a sensory data fusion strategy is used to extract pertinent information from the multiple sensory data. The second control approach consists of selecting, among the available sensory signals, a set of pertinent data, which is then servoed. The two approaches are referred to as sensory data fusion and sensory data selection.

The International Journal of Robotics Research
Vol. 20, No. 5, May 2001, pp. 364-377,

A typical example of sensory data fusion is stereo vision. With this approach, two images provided by two distinct cameras are used to extract complete Euclidean information about the observed scene. The fused data can then be used as a measure in a feedback loop, such as in Allen et al. (1993), where a stereo pair was used to reconstruct the position of a moving object at video rate (one image every 0.004 seconds) and to control the robot's end effector, in order to track and select the object. Such a three-dimensional visual servoing technique was proposed in Cipolla and Hollinghurst (1997), where a five-degree-of-freedom robot was controlled using a fixed external stereo head. Data fusion can also be performed in projective space without explicitly reconstructing the position of observed objects. For example, in Hager (1997) a stereo system simultaneously tracked the robot end effector and visual features used to define goal positions. Thus, the error was defined as a function of features directly observed from both cameras, and the robot could be positioned with an accuracy that was independent of errors in hand-eye calibration. Another method, based on the estimation of a $(4 \times 4)$ projective homography matrix that relates two sets of points of the projective space, was proposed in Ruf and Horaud (1999). This method is similar to the previous methods, but the robot is controlled without performing any Euclidean reconstruction. Sensory data fusion can be achieved directly by means of control, as in Maru et al. (1993), where two-dimensional visual servoing was performed using a stereo head while updating the task Jacobian from the information provided by the two cameras. Disparate sensory data fusion can also be used in robot control. For example, in Agapakis (1990) a proximity sensor was used in combination with a camera for depth reconstruction purposes.

Sensory data selection is used when the different data do not provide information of the same quality. In this case, one can use environment models to select the appropriate sensor and to switch control between sensors. A typical example of this approach was developed in Nelson and Khosla (1996) and Nelson, Morrow, and Khosla (1996) using the resolvability concept, which provides a measure of the ability of a sensor to resolve motion. The authors considered an application involving a combination of force and vision. The resolvability of the two sensors was evaluated on-line, and the controller automatically switched between force and vision control during a contact task. A similar idea was developed in Hosoda, Igarashi, and Asada (1996) by applying classical hybrid position/force control. The force was servoed along the direction that was mechanically constrained by the environment (e.g., the normal direction of a contact plane) while the image was used to servo the robot's motion along the remaining free directions. The sensory signal selection was operated in advance.

Our approach to multisensory robot control is drastically different from these two methodologies (Morel, Malis, and Boudet 1998; Malis, Chaumette, and Boudet 2000). It does not pertain to sensory data fusion, since we assume that the sensors may observe different physical phenomena from which the extraction of unique information does not make sense. Nor does it pertain to sensory data selection, since we consider potential situations for which it is not possible to select a set of data that would be more pertinent than others. Consequently, the proposed approach addresses a very large spectrum of potential applications for which the sensory equipment may be disparate and complex. As an improvement over previous approaches, there is no need to provide a model of the environment that would be required to design a switching or fusion strategy.

Consider an example task in nuclear maintenance operations consisting of placing the nozzle dam in steam generators (see Fig. 1). The object is a disk of roughly 1 m diameter that must overlap a hole where it fits precisely. Because in typical maintenance scenarios the exact location of the hole with respect to the robot is not known precisely, vision feedback is necessary for an automatic operation. Due to the large size of the object, a single camera observing only one side of the scene is not appropriate, since it could produce a "lever effect." A small positioning error of the camera could produce a larger error on the other side of the cover. The use of more cameras allows one to observe different parts of the scene. The cameras can be placed on the robot around the cover to observe the edges of the hole, as illustrated in Figure 1. In this case, the fields of view of the cameras do not necessarily overlap, in contrast to conventional stereo vision. Furthermore, in the final phase of the task, calibration errors in the vision system may cause small displacements from the nominal insertion trajectory. This would involve large forces. A wrist force/torque sensor can then be used in conjunction with the cameras to perform the placement task while minimizing

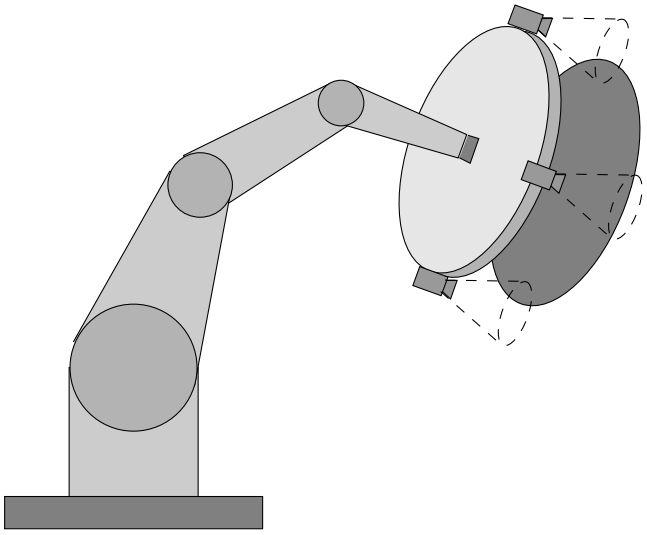


Fig. 1. Cover positioning in a steam generator.

interaction forces. It must be noted that combining force and vision has rarely been considered, and to our knowledge only a few references are available in the literature (Hosoda, Igarashi, and Asada 1996; Nelson and Khosla 1996; Nelson, Morrow, and Khosla 1996; Morel, Malis, and Boudet 1998).

The main contribution of our work is the definition of a global task function as the weighted sum of individual task functions (one for each sensor). Such a hierarchical structure leads to an increased modularity. A task function for each subsystem is designed, and the sensor combination is performed at a higher level.

The paper is organized as follows. In the next section, a general formulation of the disparate multisensor control problem is provided based on the task function concept (Samson, Le Borgne, and Espiau 1991). The general control law and the stability analysis of the controller are described in Section 3. In Section 4, we detail the design of the force and vision control subsystems. Finally, experiments are presented in Section 5.

## 2. Modeling

In this section, we describe the general model used to control the robot motion from disparate sensory feedback. The robot is supposed to be controlled by a six-dimensional vector $\mathbf{v}_r$ representing the end effector velocity whose components are supposed to be expressed in the end effector frame. It is equipped with $N$ sensors, each rigidly attached to the end effector. Each sensor $S_i$ $(i = 1 \cdots N)$ provides an $n_i$-dimensional vector signal $\mathbf{s}_i$. For simplicity, it is assumed that $n_i \geq 6$. However, this hypothesis is not restrictive, and our work can be easily extended to the general case. An interaction matrix $\mathbf{L}_i$ $\{i = 1, 2, .., N\}$ is attached to each sensor such that (Espiau, Chaumette, and Rives 1992)

$$\dot{\mathbf{s}}_i = \mathbf{L}_i \mathbf{v}_i, \tag{1}$$

where $\mathbf{v}_i$ is the absolute velocity of the sensor $S_i$ expressed in a frame $\mathcal{F}_i$ attached to the sensor, where $m$ is the number of degrees of freedom of the manipulator (generally $m = 6$). Furthermore, we introduce the transformation matrix $\mathbf{W}_i$ linking the sensor velocity and the end effector velocity:

$$\mathbf{v}_i = \mathbf{W}_i \mathbf{v}_r. \tag{2}$$

Because the sensor is fixed with respect to the end effector, the matrix $\mathbf{W}_i$ is constant and depends on the geometric transform between the end effector frame and the sensor frame. Using the two previous equations, we get $\dot{\mathbf{s}}_i = \mathbf{L}_i \mathbf{W}_i \mathbf{v}_r$, where $\mathbf{L}_i \mathbf{W}_i$ is called the Jacobian matrix of the $i$th sensor.

Let $\mathbf{s} = \left[\mathbf{s}_1^T \; \mathbf{s}_2^T \; \cdots \mathbf{s}_N^T\right]^T$ be the $n$-dimensional vector (where $n = \sum_{i=1}^{N} n_i$) containing all the signals provided by the $N$ sensors. The relationship between the time derivative of the global sensor signal vector and the end effector velocity $\mathbf{v}_r$ is

$$\dot{\mathbf{s}} = \begin{bmatrix} \mathbf{L}_1 & 0 & \dots & 0 \\ 0 & \mathbf{L}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{L}_N \end{bmatrix} \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \\ \vdots \\ \mathbf{W}_N \end{bmatrix} \mathbf{v}_r = \mathbf{L}\mathbf{W}\mathbf{v}_r. \tag{3}$$

Now, let $\mathbf{s}^*$ be the desired value of the sensor signal vector $\mathbf{s}$. We use a task function of the form $\mathbf{e} = \mathbf{C}(\mathbf{s} - \mathbf{s}^*)$ (Espiau, Chaumette, and Rives 1992), where $\mathbf{C}$ is a full-rank constant matrix of dimensions $(6 \times n)$, which allows one to take into account the information redundancy. The matrix $\mathbf{C}$ being constant, the time derivative of the task function is

$$\dot{\mathbf{e}} = \mathbf{C}\dot{\mathbf{s}} = \mathbf{C}\mathbf{L}\mathbf{W}\mathbf{v}_r. \tag{4}$$

A major concern in designing a task function–based controller is to select a suitable constant matrix $\mathbf{C}$ while ensuring that the matrix $\mathbf{CLW}$ has full rank. Usually, $\mathbf{C}$ is designed as a function of the matrices $\mathbf{L}$ and $\mathbf{W}$, which both depend on unknown parameters. Thus, estimated matrices $\widehat{\mathbf{L}}$ and $\widehat{\mathbf{W}}$ will be used. Different choices of $\mathbf{C}$ are possible. Mainly, two solutions involving a constant matrix $\mathbf{C}$ are used in the literature:

- $\mathbf{C} = (\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^+$ is the pseudoinverse of the Jacobian matrix calculated at the desired position $\widehat{\mathbf{L}}^* = \mathbf{L}(\mathbf{s}^*, \mathbf{g}^*)$ (where $\mathbf{g}$ are unknown geometric parameters). This choice ensures a better decoupling of the control law near the convergence.
- $\mathbf{C} = (\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^T$ is the transpose of the Jacobian matrix calculated at the desired position. This choice does not need the inversion of the interaction matrix, but it does not realize the decoupling of the control law near the convergence.

In this paper, we propose to use the block pseudoinverse of the Jacobian matrix:

$$\mathbf{C} = (\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^\sharp = \left[ \begin{array}{ccc} \kappa_1 \widehat{\mathbf{W}}_1^{-1} \widehat{\mathbf{L}}_1^{*+} & \dots & \kappa_N \widehat{\mathbf{W}}_N^{-1} \widehat{\mathbf{L}}_N^{*+} \end{array} \right], \tag{5}$$

where $\widehat{\mathbf{L}}_i^{*+}$ is the pseudoinverse of $\widehat{\mathbf{L}}_i^*$, $\widehat{\mathbf{W}}_i^{-1}$ is the inverse of $\widehat{\mathbf{W}}_i$, and $\kappa_i$ is a positive weighting factor such that $\sum_{i=1}^{N} \kappa_i = 1$. The choice of C made in eq. (5) presents intermediate characteristics with respect to the choices described above (in the items).

The usefulness of this new combination matrix for disparate multiple sensory feedback will be shown when computing the control law and analyzing the stability of the system. However, it can already be noted that it has some interesting characteristics. Indeed, if we consider for each sensor a task function $\mathbf{e}_i = \mathbf{C}_i \left(\mathbf{s}_i - \mathbf{s}_i^*\right)$, where $\mathbf{C}_i = \widehat{\mathbf{W}}_i^{-1} \widehat{\mathbf{L}}_i^{*+}$, then the task function of the entire system is a weighted sum of the task functions relative to each sensor:

$$\mathbf{e} = \mathbf{C}\left(\mathbf{s} - \mathbf{s}^*\right) = \sum_{i=1}^{N} \kappa_i \mathbf{e}_i = \sum_{i=1}^{N} \kappa_i \mathbf{C_i} \left(\mathbf{s_i} - \mathbf{s_i^*}\right). \tag{6}$$

Such a choice of the matrix $\mathbf{C}$ leads to an increased modularity. An individual task function is designed for each subsystem, and the sensor combination is performed at a higher level. The design of the multisensor combination simply consists of selecting the positive weights $\kappa_i$ ($i = 1 \cdots N$). This choice is both task and sensor dependent. The weights $\kappa_i$ can be set according to the relative precision of the sensors or, more generally, to balance the velocity contribution of each sensor. If a sensor $S_i$ plays an important safety role (e.g., a force sensor), the corresponding weight $\kappa_i$ will be large. A dynamical setting of $\kappa_i$ can also be implemented. For example, if a sensor $S_i$ fails during a task, then the weight $\kappa_i$ will be set to zero.

## 3. Control Law and Stability Analysis

If a simple proportional control law is set to work (e.g., by imposing the exponential decreasing of the task function $\dot{\mathbf{e}} = -\lambda\mathbf{e}$, where $\lambda$ is a positive scalar), then the ideally decoupling control law is

$$\mathbf{v}_r = -\lambda(\mathbf{CLW})^{-1}\mathbf{e}. \tag{7}$$

A more realistic control law is the following:

$$\mathbf{v}_r = -\lambda(\mathbf{C}\widehat{\mathbf{L}}\widehat{\mathbf{W}})^{-1}\widehat{\mathbf{e}}. \tag{8}$$

In general, for use in the control law, we choose the interaction matrix calculated at the desired position: $\widehat{\mathbf{L}} = \widehat{\mathbf{L}}^*(\mathbf{s}^*, \widehat{\mathbf{g}}^*)$. Consequently, choosing $\mathbf{C} = (\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^\sharp$ defined by equation (5), we obtain

$$(\mathbf{C}\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^{-1} = \left(\sum_{i=1}^{N} \kappa_i \widehat{\mathbf{W}}_i^{-1}\widehat{\mathbf{L}}_i^{*+}\widehat{\mathbf{L}}_i^*\widehat{\mathbf{W}}_i\right)^+ = \sum_{i=1}^{N} \kappa_i \mathbf{I}_m = \mathbf{I}_m, \tag{9}$$

and the control law is

$$\mathbf{v}_r = -\lambda\widehat{\mathbf{e}}. \tag{10}$$

Similar results can be obtained with different choices of $\widehat{\mathbf{L}}$, as will be seen in the case of the force subsystem. Considering eq. (6), this control law is equivalent to a weighted sum of the control laws of each subsystem $\mathbf{v}_{ri} = -\lambda\widehat{\mathbf{e}}_i$:

$$\mathbf{v}_r = -\lambda\sum_{i=1}^{N} \kappa_i\widehat{\mathbf{e}}_i = \sum_{i=1}^{N} \kappa_i\mathbf{v}_{ri}. \tag{11}$$

The general structure of the controller is shown in Figure 2. To obtain the closed-loop system equation, the estimated task function must be expressed as a function of the real task function and of the calibration errors of the system. In the general case, when the task function is built directly from sensor data, it is simply $\widehat{\mathbf{e}} = \mathbf{e}$ if the measurement noise is neglected. When the task function is built using parameters reconstructed from the sensor data, a further modeling step should be performed, as will be seen in the next section. If $\widehat{\mathbf{e}} = \mathbf{e}$, the closed-loop system with the control law (10) is given by

$$\dot{\mathbf{e}} = -\lambda(\mathbf{CLW})(\mathbf{C}\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^{-1}\mathbf{e} = -\lambda(\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^\sharp(\mathbf{LW})\mathbf{e}. \tag{12}$$

A sufficient condition for the stability of the system is

$$(\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^\sharp\mathbf{LW} > 0. \tag{13}$$

It is clear now why it is interesting to choose $\mathbf{C}$ as a block pseudoinverse. Indeed, using eq. (5), the condition (13) can also be written as

$$(\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^\sharp\mathbf{LW} = \sum_{i=1}^{N} \kappa_i \widehat{\mathbf{W}}_i^{-1}\widehat{\mathbf{L}}_i^{*+}\mathbf{L}_i\mathbf{W}_i > 0, \tag{14}$$

and this condition is verified if (remember that $\kappa_i > 0$)

$$\widehat{\mathbf{W}}_i^{-1}\widehat{\mathbf{L}}_i^{*+}\mathbf{L}_i\mathbf{W}_i > 0 \qquad \forall i = \{1, 2, 3, ..., N\}. \tag{15}$$

Thus, the whole system will be stable if each subsystem is stable, since the sum of positive matrices is a positive matrix. The stability analysis can thus be reduced to the study of the stability of each subsystem separately. Even if the system is not unstable (i.e., $\|\widehat{\mathbf{e}}\|$ is bounded and cannot grow indefinitely), it could happen that $\widehat{\mathbf{e}} = 0$ but $\widehat{\mathbf{e}}_i \neq 0 \forall i$. In this case, the control output will be $\widehat{\mathbf{v}} = 0$, which corresponds to a local minimum. However, that situation can easily be detected, and one can choose a new set of gains $\kappa_i$ such that $\widehat{\mathbf{e}} \neq 0$.

## 4. Subsystem Design

In this section, we discuss the design of each separate subsystem. The force and vision control subsystems described in this section can be combined in the general control scheme described earlier.

### *4.1. Vision Subsystem*

Vision-based control can be achieved using three different methods. In a three-dimensional visual servoing system, the error to be controlled corresponds to the camera's pose, that is, its position and orientation (Wilson, Hulls, and Bell 1996). The pose relative to the target is estimated from image features, which requires the precise knowledge of the target geometry. Conversely, two-dimensional visual servoing exploits an error directly computed from the image features relative to their desired values (Espiau, Chaumette, and Rives 1992). Finally, in the two-and-a-half-dimensional visual servoing approach, the error to be controlled is computed in part in the Cartesian space and in part directly in the image (Malis, Chaumette, and Boudet 1999; Morel et al. 1999). Any of the three methods can be used in our framework. However, the three-dimensional servoing approach is not considered in this paper.

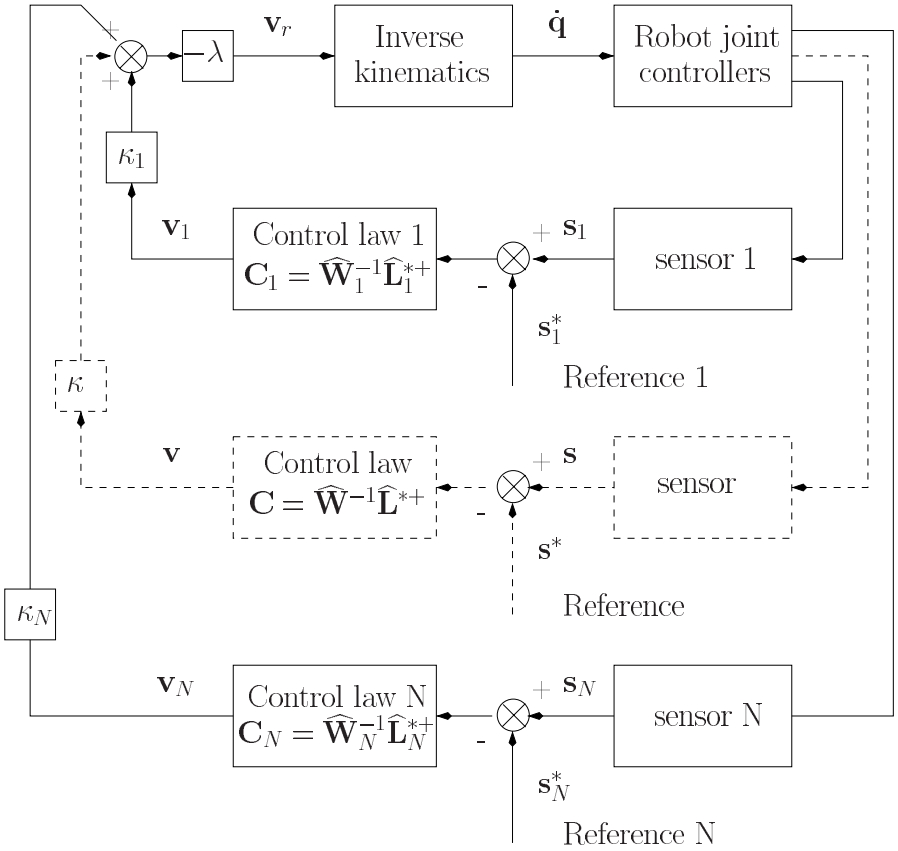



Fig. 2. General structure of the controller with $N$ sensors.

#### *4.1.1. Two-Dimensional Visual Servoing*

In the implemented image-based controller, the considered image features are the image coordinates of $p$ points observed on the target $\mathbf{s} = \begin{bmatrix} x_1 & y_1 & \dots & x_p & y_p \end{bmatrix}^T$. The task function is $\mathbf{e} = \mathbf{C}(\mathbf{s} - \mathbf{s}^*)$. Thus, the Jacobian matrix of the sensor signal is

$$\frac{\partial \mathbf{s}}{\partial \mathbf{r}} = \mathbf{L}(\mathbf{z}, \mathbf{s})\mathbf{W}. \tag{16}$$

The interaction matrix $\mathbf{L}$ depends on the distance $\mathbf{z}$ between the points of the target and the camera. When an object is in the field of view of at least two cameras (stereo configuration), the distance can be estimated. Otherwise, it can be approximated for each camera, exactly as for mono-camera visual servoing (Espiau, Chaumette, and Rives 1992), when the desired features in the images $\mathbf{s}^*$ are stored. The optimal choice for the combination matrix is $\mathbf{C} = \widehat{\mathbf{W}}^{-1}\widehat{\mathbf{L}}^{+}(\widehat{\mathbf{z}}^*, \widehat{\mathbf{s}}^*)$. The vision system is stable if

$$\widehat{\mathbf{W}}^{-1}\widehat{\mathbf{L}}^{+}(\widehat{\mathbf{z}}^*, \widehat{\mathbf{s}}^*)\mathbf{L}(\mathbf{z}, \mathbf{s})\mathbf{W} > 0. \tag{17}$$

Unfortunately, the stability domain of the two-dimensional visual servoing is not analytically known to this date, and the analysis has to be stopped. However, a two-dimensional mono-camera visual servoing system is known to be robust with regard to calibration errors (Espiau 1993). The same robustness can thus be expected from a multicamera system using a two-dimensional visual servoing technique.

#### *4.1.2. Two-and-a-Half-Dimensional Visual Servoing*

In the case of two-and-a-half-dimensional visual servoing, the task function is reconstructed from the sensor signal (Malis, Chaumette, and Boudet 1999). The Jacobian matrix is an upper block-triangular matrix that is nonsingular in the whole workspace:

$$\frac{\partial \mathbf{s}}{\partial \mathbf{r}} = \mathbf{L}(d^*, \mathbf{s})\mathbf{W}. \tag{18}$$

The only unknown parameter is the distance $d^*$ between the camera and the target, but it does not have a great influence on the stability of the system. In that case, we have $\widehat{\mathbf{e}}_i = \mathbf{E}_i\mathbf{e}_i$, where $\mathbf{E}_i$ depends on the camera's internal parameters and is given in Malis, Chaumette, and Boudet (1999). The estimated task function is thus given by

$$\widehat{\mathbf{e}} = \sum_{i=1}^{N} \kappa_i \mathbf{E}_i \mathbf{e}_i. \tag{19}$$

Consequently, $\widehat{\mathbf{e}}$ cannot be easily written as a function of $\mathbf{e}$ under the form $\widehat{\mathbf{e}} = \mathbf{E}\ \mathbf{e}$. However, we can proceed in the following way. The task function can be written as

$$\mathbf{e} = \left[ \begin{array}{cccc} \kappa_1 \mathbf{I}_6 & \kappa_2 \mathbf{I}_6 & \ldots & \kappa_N \mathbf{I}_6 \end{array} \right] \mathbf{e}^{'} = \mathbf{K}\mathbf{e}^{'}, \tag{20}$$

where $\mathbf{e}^{'}$ is the $6N$-dimensional vector containing the task functions of each subsystem. This vector can be computed as

$$\mathbf{e}^{'} = \mathbf{K}^{+}\mathbf{e} + (\mathbf{I}_{6N} - \mathbf{K}^{+}\mathbf{K})\phi, \tag{21}$$

where $(\mathbf{I}_{6N} - \mathbf{K}^{+}\mathbf{K})\phi$ is an arbitrary vector belonging to the null space of $\mathbf{K}$. Because of the simple form of $\mathbf{K}$, its pseudoinverse is $\mathbf{K}^{+} = (\sum_{i=1}^{N} \kappa_i^2)^{-1}\mathbf{K}^{T}$. Plugging eqs. (21) and (20) in eq. (19), the estimated task function is

$$\widehat{\mathbf{e}} = \frac{1}{\sum_{i=1}^{N} \kappa_i^2} \sum_{i=1}^{N} \kappa_i^2 \mathbf{E}_i\ \mathbf{e} + \mathbf{b}_e = \mathbf{E}\ \mathbf{e} + \mathbf{b}_e, \tag{22}$$

where $\mathbf{b}_e$, which can be considered as an additional "bias" on the task function, is given by

$$\mathbf{b}_e = \sum_{i=1}^{N} \kappa_i \mathbf{E}_i \phi_i - \frac{1}{\sum_{i=1}^{N} \kappa_i^2} \sum_{i=1}^{N} \kappa_i^2 \mathbf{E}_i \left( \sum_{i=1}^{N} \kappa_i \phi_i \right). \tag{23}$$

This bias can be neglected. Indeed, this hypothesis is justified because, owing to the simple form of matrix $\mathbf{E}_i$ (block diagonal) and with an appropriate choice of $\mathbf{e}_i$ for the rotation ($\mathbf{e}_{\omega i} = \widehat{\mathbf{A}}\mathbf{A}^{-1}\mathbf{u}\theta$, where $\mathbf{A}$ is the matrix of the camera parameters, $\widehat{\mathbf{A}}$ is its approximation, and $\mathbf{u}\theta$ is the rotation vector), $\mathbf{E}_i \approx \mathbf{I}_6$ and $\mathbf{b}_e \approx 0$. Let us note that at the convergence, $\mathbf{E}_i = \mathbf{I}_6$ and $\mathbf{b}_e = 0$ and the system is always locally stable (Malis 1998). Furthermore, because of the simple form of matrices $\mathbf{L}_i$ and $\mathbf{W}_i$ (block triangular), the stability analysis of the multicamera two-and-a-half-dimensional visual servoing is thus reduced to the stability analysis of each subsystem. Because each subsystem is stable under the conditions given in Malis, Chaumette, and Boudet (1999), the whole system is also stable, at least under the same conditions.

### *4.2. Force Subsystem*

To design the force subsystem, it is straightforward to choose $\mathbf{s} = \boldsymbol{\tau}$, where $\boldsymbol{\tau}$ is the interaction wrench. The task function is again $\mathbf{e} = \mathbf{C}(\mathbf{s} - \mathbf{s}^{*})$. Then, the interaction matrix is

$$\frac{\partial \mathbf{s}}{\partial \mathbf{r}} = \frac{\partial \boldsymbol{\tau}}{\partial \mathbf{r}}, \tag{24}$$

where $\mathbf{r}$ is the actual end effector pose. The interaction matrix depends on both the geometry of the constraint and the contact mechanics. Thus, a general stability condition cannot be derived for any task. We provide an example considering a contact point configuration. Thus, only linear forces $\tau$ are considered in the example instead of the complete wrench $\boldsymbol{\tau}$. In this case, the desired damping is a constant spherical matrix, $\mathbf{B} = b\mathbf{I}_3$. If we assume elastic deformations and no friction, we obtain (Espiau, Merlet, and Samson 1990)

$$\frac{\partial \tau}{\partial \mathbf{r}} = k\mathbf{n}\mathbf{n}^{T}, \tag{25}$$

where $\mathbf{n}$ is the unitary vector normal to the contact surface and $k$ is the scalar stiffness. Choosing $\mathbf{C} = \mathbf{B}^{-1}$ and the control law $\dot{\mathbf{r}} = -\lambda\mathbf{e}$, the system will be stable if

$$\mathbf{B}^{-1}\frac{\partial \boldsymbol{\tau}}{\partial \mathbf{r}} > 0. \tag{26}$$

Because the desired damping is a constant spherical matrix, $\mathbf{B}^{-1} = \frac{1}{b}\mathbf{I}_3$, we finally obtain

$$\mathbf{B}^{-1}\frac{\partial \boldsymbol{\tau}}{\partial \mathbf{r}} = \frac{k}{b}\mathbf{n}\mathbf{n}^{T}, \tag{27}$$

which is positive for any positive value of $b$ and $k$. As already mentioned, this example takes into account only linear forces. Similarly, one can consider the general case with the complete wrench $\boldsymbol{\tau}$.

It is interesting to note that we finally obtain what is referred to in the force control literature as a position-based impedance controller (Heinrichs, Sepehri, and Thornton-Trump 1996; Caccavale et al. 1999), with a desired impedance corresponding to a pure damper.

## 5. Experimental Results

The experimental setup used in this work is as follows. The robot is a seven-axis redundant electric Mitsubishi PA-10 manipulator (at the Direction des Etudes et Recherches at Electricité de France). The joint redundancy of the robot is solved by minimizing joint velocities in the inverse kinematics procedure using a conventional pseudoinverse algorithm $\dot{\mathbf{q}} = \mathbf{J}^{+}\mathbf{v}_r$. In addition, a second-order differential model is used to solve the inverse kinematics in singular joint configurations (Malis, Morin, and Boudet 1996). We used Panasonic cameras and an Assurance Technologies Inc. six-axis force/ torque sensor mounted at the end effector. Low-level joint position control is achieved via the Mitsubishi controller, which communicates with a VERSAmodule Eurocard bus controller through an Arcnet communication link. Two CPU boards supporting VxWorks realize the position-based impedance control. An additive specific EDIXIA vision-dedicated board generates the reference velocity.

To perform a task, it is assumed that an off-line learning phase was performed prior to the maintenance operation. The off-line learning phase provides the final desired sensory signal value, which consists of the images viewed by the different cameras at the final location. If the end effector positioning requires an operating force to be applied, the desired force

signal can be taught to the system during the same learning phase. Finally, the task consists of servoing the different sensory signals toward their final desired value, assuming that sufficient information is available at the initial stage of the task, that is, at least one camera can see its tracked target. We describe experiments in which two cameras are coupled and experiments in which a camera and a force sensor are coupled.

### *5.1. Coupling Two Cameras*

A system simulating the positioning of the cover using only vision was tested. To demonstrate the practicability of the task, only two cameras were used. The cameras are placed at the extremity of a 50-cm aluminum bar mounted on the manipulator robot as in Figure 3. Two planar targets with six points each were used. A simple hardware architecture was used in order to reduce the cost of the system. The two cameras are connected to two different entries of the same EDIXIA video board, and the acquisition of the two images is sequential. The rate of such a system is two times as slow as the rate of a parallel system (i.e., 80 ms, with a delay between the acquisition of the two images of about 40 ms). Consequently, the velocity of the robot end effector during the servoing cannot be too high. Indeed, the delay between the acquisitions of the images is not taken into account in the control law. The convergence speed of the task was set to $\lambda = 0.1$ for all the experiments. Depending on the stability domain that can be obtained from the control law, two types of algorithms can be considered. When the convergence domain of each subsystem is not large (as for the two-dimensional visual servoing), the following general algorithm is proposed (using $N$ cameras):

1. Test how many cameras have converged (a camera has converged if the error on visual features is lower than $\epsilon_i$: $\|\mathbf{s}_i - \mathbf{s}_i^*\| < \epsilon_i$). If all the cameras have converged, the goal is reached. Go to 2.
2. Select all the converged cameras plus the camera nearest to convergence (the camera nearest to convergence is the one minimizing $\|\mathbf{s}_j - \mathbf{s}_j^*\|$ under the constraint that $\|\mathbf{s}_j - \mathbf{s}_j^*\| > \epsilon_j$). Go to 3.
3. Perform visual servoing using the selected cameras. After their convergence, go to 1.

This iterative algorithm adds a new camera at each step. If the stability domain is very large (as for two-and-a-half-dimensional visual servoing), a more satisfactory algorithm can be considered:

1. Test how many cameras have converged. If all have converged, the goal is reached. Go to 2.
2. Select all the converged cameras plus those that have the target in their field of view (and are in the stability domain). Go to 3.

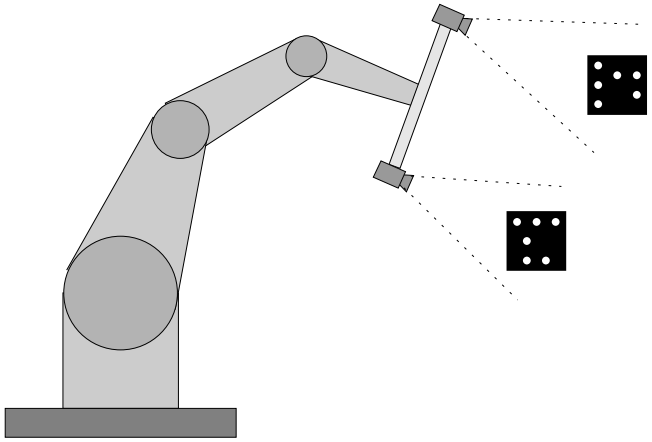


Fig. 3. Positioning a cumbersome object.

3. Perform visual servoing using the selected cameras. After their convergence, go to 1.

If the cameras are well distributed around the object, then these two algorithms converge at least in $N$ steps. This implies the following:

- A converged camera has also converged at the end of each step. This does not mean that the camera stays at convergence during the servoing. Indeed, only the whole task function decreases.
- If $p$ (with $p < N$) cameras have converged, there exists at least one other "active" camera, that is, with the target in its field of view and in the stability domain.

#### *5.1.1. Precision Comparison between Mono-Camera and Multicamera Visual Servoing*

In these experiments, we used image-based visual servoing. As a result of the use of a teaching-by-showing technique, the system is robust with regard to the calibration errors. The system is first shown the target views. The image features are automatically extracted and stored. The robot and object are then moved, and the system automatically identifies the target and controls the robot end effector. The visual servoing is stopped when the maximal error on the coordinates of the image points is 1 pixel. Four experiments were carried out corresponding to four different camera displacements. The first displacement was a pure translation. The second and the third were mainly rotations around the $\vec{x}$ and the $\vec{y}$ axes, respectively. Finally, the fourth displacement was a large rotation around the $\vec{x}$ and $\vec{y}$ axes and a large translation. First, the visual servoing is carried out using only one of the two cameras ($\kappa_1 = 0$ and $\kappa_2 = 1$). Then, starting from the same initial position, the servoing is performed using the two cameras with the control law (10) ($\kappa_1 = 0.5$ and $\kappa_2 = 0.5$). In all the experiments, the positioning accuracy (i.e., the

**Table 1. Positioning Accuracy**

| Position | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Translation | 1.7 mm, 31 % | 2.0 mm, 56 % | 1.9 mm, 46 % | 2.0 mm, 48 % |
| Rotation | 0.3 degrees, 36 % | 0.4 degrees, 58 % | 0.3 degrees, 27 % | 0.4 degrees, 44 % |

NOTE: The absolute number is the positioning error obtained using one camera. The percentage is the improvement of the positioning using two cameras.

error between the reached position and the desired position) is improved considerably with the use of multicamera visual servoing, as is shown in Table 1 (typically from 2 mm to 0.9 mm). These results demonstrate the feasibility of using several cameras to position a cumbersome object.

#### *5.1.2. Comparison of Two-Dimensional and Two-and-a-Half-Dimensional Visual Servoing*

Even though multicamera visual servoing can be used with any scheme (two-dimensional, two-and-a-half-dimensional, and three-dimensional), we compare only the two-dimensional and two-and-a-half-dimensional methods. The results are illustrated in Figure 4 and Figure 5, respectively. Each figure is divided into two columns, one for each camera. In each column, we provide the error on the coordinates of the points versus the number of iterations and their trajectory in the image (diamond and circle marks correspond to the initial and final positions, respectively). In the last row, we provide the control law (i.e., the velocity of the robot end effector) versus the number of iterations. Even with the imprecise calibration used, the error on the coordinates goes to zero and the control law is stable. Indeed, the camera's intrinsic parameters and the transformation matrices between each camera and end effector frame were only roughly known.

As a result of the use of a simple proportional control law, the speed of convergence is relatively slow at the end of the servoing. It can be easily improved by increasing gain $\lambda$ as soon as the error becomes small. Obviously, increasing $\lambda$ too much will lead to an oscillatory behavior. In our case, $\lambda$ is chosen small, since the rate of the vision system is 80 ms and the velocity of the end effector must be limited. For both schemes, the trajectories of the points in the images are the same (even if the control law is different), since the displacement was not very large. Experiments with larger displacements were also realized, but only one target was initially in the image. In this case, we first perform the visual servoing using only one camera. Once the two targets are in the images, we perform multicamera visual servoing.

### *5.2. Coupling Vision and Force Sensors*

The illustrative task of coupling vision and force sensors is part of an actual nuclear power plant valve maintenance operation. To be able to use the different tools involved in this task (see Fig. 6a), a female interface is mounted on each tool, which matches the end effector–mounted electro-pneumatic male tool changer.

The clearance is less than a 10th of a millimeter, and the tool location is unknown. A detailed view of the tool changer mounted on the robot's end effector is provided in Figure 6b. A system to realize the insertion of a tool changer (which is a task equivalent to the insertion of the cover) using vision and force sensors was tested. Associated with our controller, a simple programming methodology was developed. It uses two steps:

- The task is run "manually"—that is, only the force subsystem control is running ($\kappa_1 = 1$ and $\kappa_2 = 0$)—and the reference trajectory is provided by a device in the place of the vision servo controller.
- Once the desired final position is reached and the force feedback loop is stabilized to zero ($\tau^* = s^* = 0$), the desired image features are computed and stored.

To be able to reach this position through combined visual and force control, one simply sets the desired image features to their memorized value while the desired force is set to zero. Thus, programming a task as complex as tool changer insertion is extremely simple.

To illustrate the experimental behavior of our approach, we compare two tool changer insertion attempts. The first is done with pure visual servoing ($\kappa_1 = 0$ and $\kappa_2 = 1$), whereas the second uses combined force and vision control ($\kappa_1 = 0.5$ and $\kappa_2 = 0.5$). The rate of the vision control is 80 ms, whereas the rate of the force control is 10 ms. Thus, the speed of the end effector is limited by the slow video rate. Improving the vision system will improve the performances of the combined control.

For the experiments shown in this paper, the target impedance is limited to pure damping $\mathbf{B}$ over the six wrench components (i.e., the components of force and torque exerted by the end effector on objects). The desired wrench is $s^* = 0$. Note that the force subsystem controller is equivalent to an accommodation control (Whitney 1977). As anticipated in Section 3, friction can significantly affect system behavior. Consider, for example, that reaching the target requires one to move the end effector parallel to the contact surface. Without friction, the vision system guides the end effector to the final position while slipping over the surface. However, in the presence of friction, the controller would fail. Close to the final position, the tangential velocity commanded by the

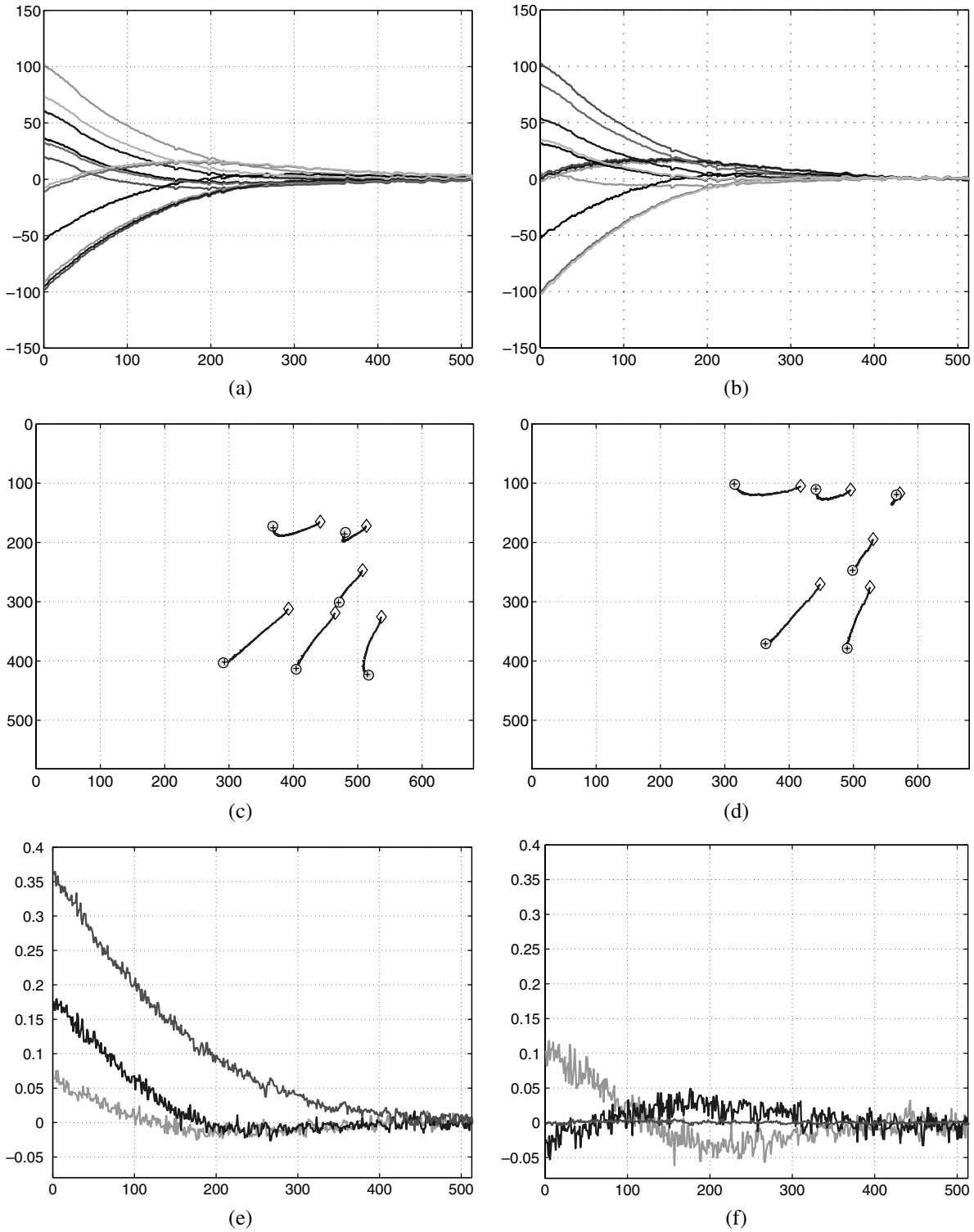



Fig. 4. Two-dimensional visual servoing results. (a) Left coordinate errors (pixels vs. iteration number). (b) Right coordinate errors (pixels vs. iteration number). (c) Left points trajectory (pixels). (d) Right points trajectory (pixels). (e) Translation velocity (cm/s vs. iteration number). (f) Rotation velocity (rad/s vs. iteration number).

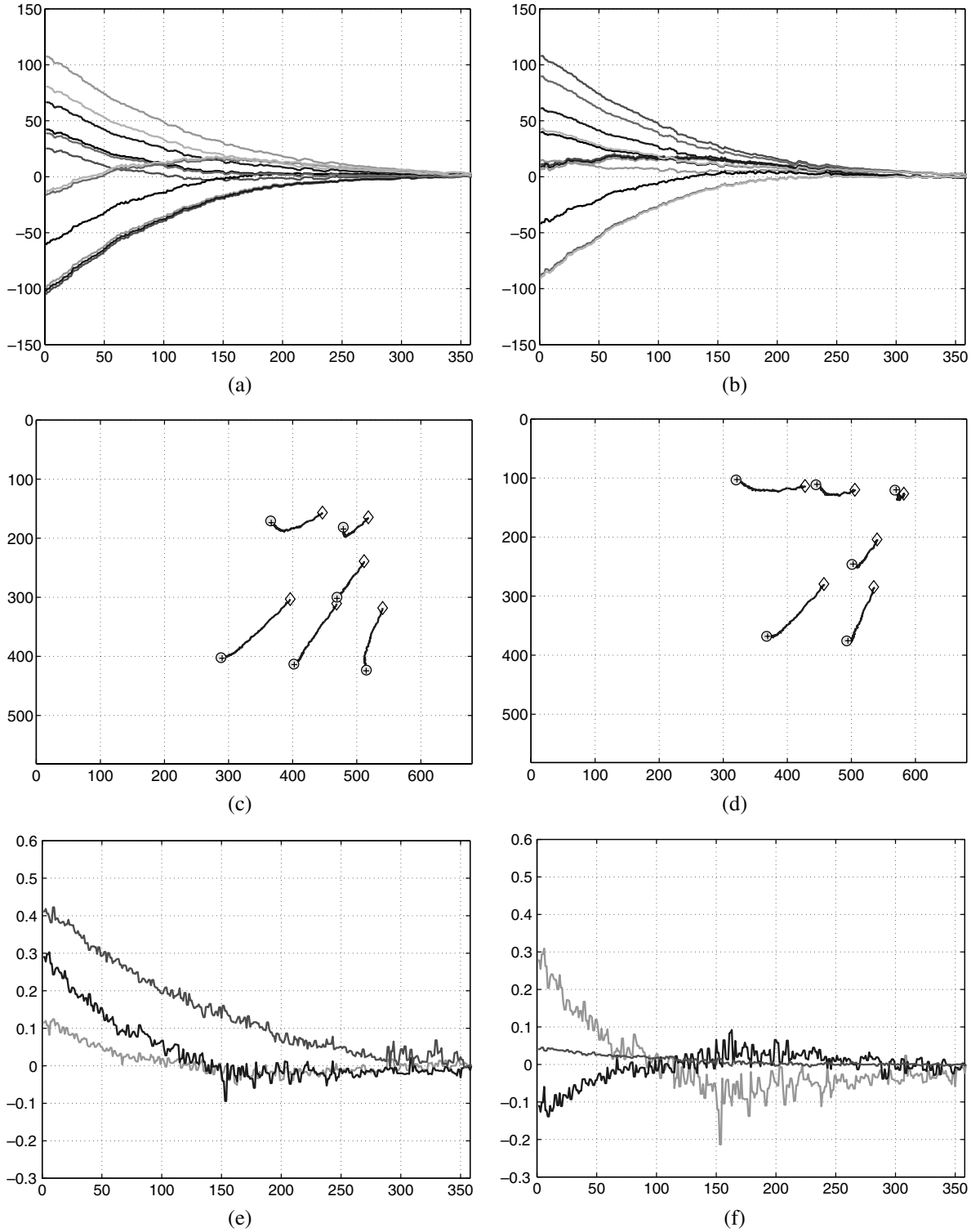
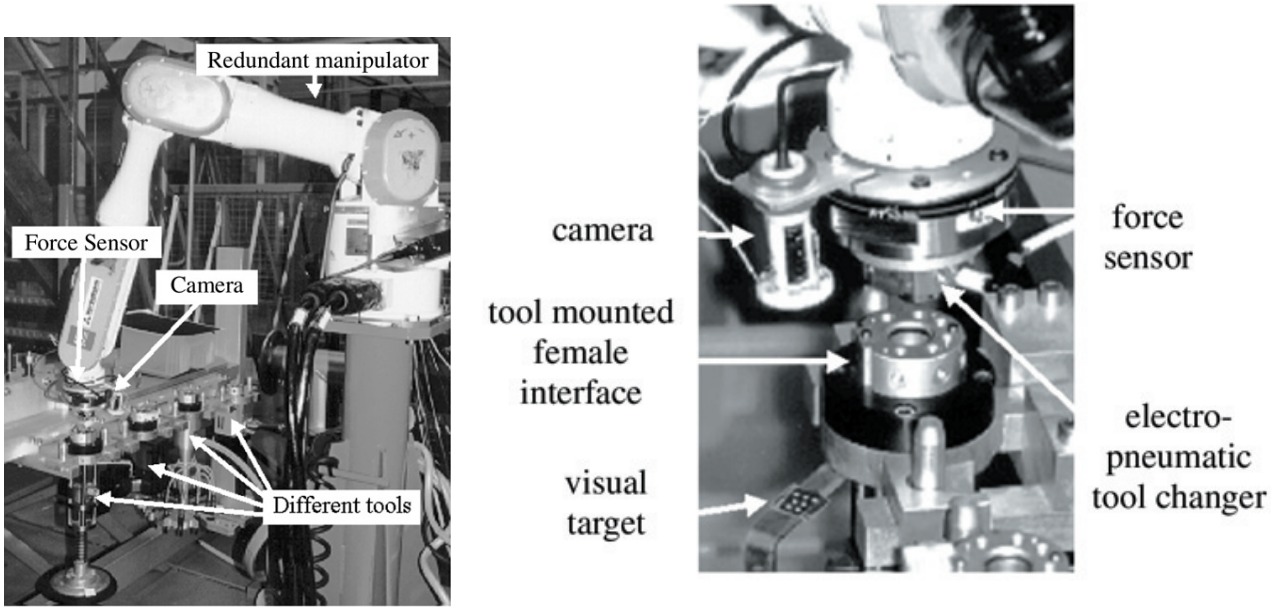



Fig. 5. Two-and-a-half-dimensional visual servoing results. (a) Left coordinate errors (pixels vs. iteration number). (b) Right coordinate errors (pixels vs. iteration number). (c) Left points trajectory (pixels). (d) Right points trajectory (pixels). (e) Translation velocity (cm/s vs. iteration number). (f) Rotation velocity (rad/s vs. iteration number).




(a) (b)

Fig. 6. Experimental setup. (a) General view of the task. (b) Detailed view of the task.

vision-based controller decreases. At some point, because of the friction component, it will become smaller than the opposite tangential velocity generated through the force feedback. With a pure damping impedance, the following condition must be respected to prevent the system from blocking:

$$|\mathbf{v}_{vision}| > |\mathbf{v}_{force}| = \mathbf{B}|\boldsymbol{\tau}|. \tag{28}$$

It is clear that with a proportional visual control law, friction generates static positioning error. A proportional integral controller can be used, but combining the integral correction with vision feedback and friction nonlinearities might lead to instability. Furthermore, limit cycles would appear. Rather, a dead zone in the force feedback loop can be used. If the force (in any direction) is smaller than a predetermined limit value, then the velocity commanded by the force loop is zero. This strategy appears to be efficient in practice.

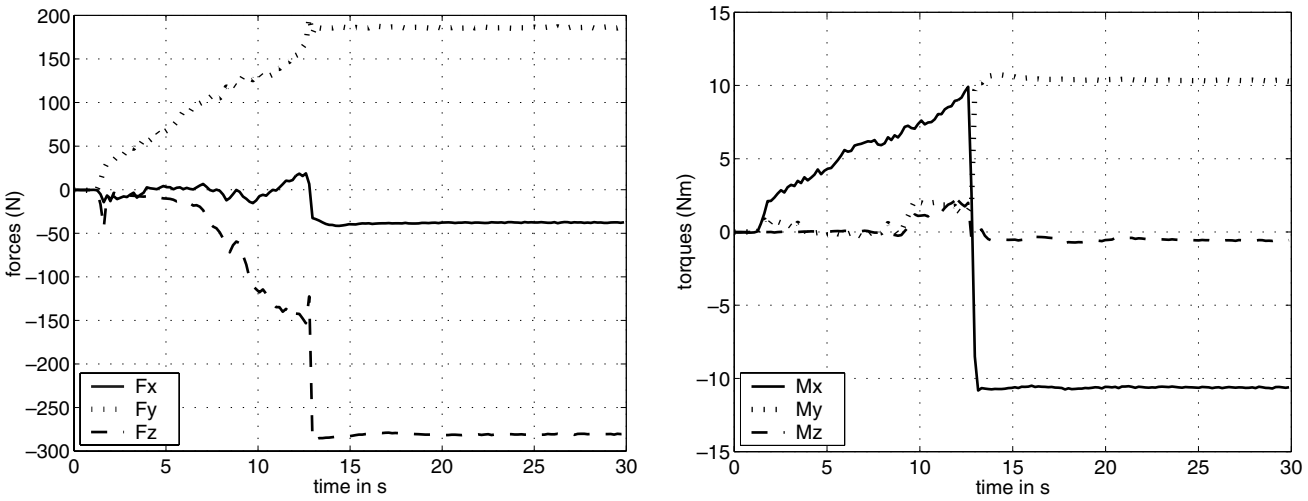
### *5.2.1. Pure Visual Servoing*

In both experiments, the initial position is reached with vision-based control and places the tool changer exactly in front of the female interface. The insertion motion is then a pure translation along the insertion axis, which corresponds to the easiest configuration. In a perfect world, the contact force would remain null. However, the vision-based control does not generate a straight-line trajectory. In fact, the trajectory is generated to minimize the image feature error, not the end effector position error. All the geometric modeling errors in both the robot and the sensory system contribute to amplify the deviation from the ideal straight-line trajectory. Thus, contact forces appear. Figure 7 shows the result for pure vision–based insertion. Owing to the integral effect of the visual control combined with the rigidity of the parts to be mated, forces and torques increase rapidly. At $t = 13$ seconds, as the end effector almost reaches the final position, the electro-pneumatic tool changer is actuated. The sudden change in contact forces is due mainly to the tool changer's clamping. Once the tool is grasped, the vision system detects final convergence and the robot stops. We note that the forces involved during the task are large. In fact, they exceed the maximum value recommended by the robot constructor.

It should be noted that the control law used for visual servoing is based only on the robot kinematics. To make our approach work properly at higher speeds (which are, however, bounded by image acquisition at video rate), it would be necessary to design the vision-based control law by taking into account the dynamics of the manipulator. However, in our experiments the inertial forces are not sufficiently large to be noticeable considering the very slow speeds due to vision control.

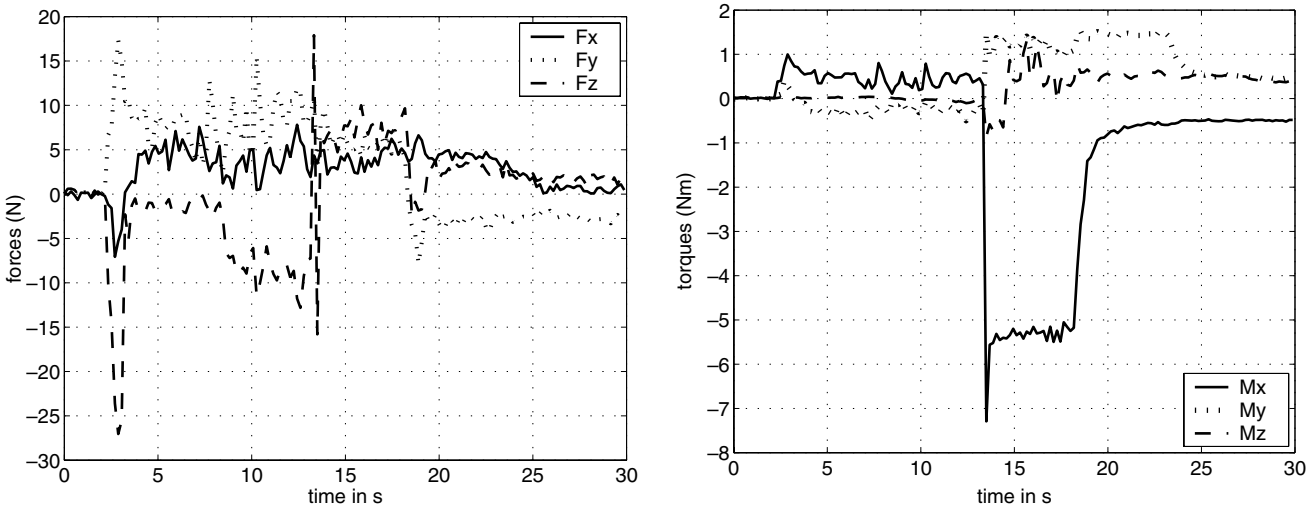
### *5.2.2. Combined Force and Vision Control*

The second experiment combines force and vision control. When the same experiment is performed with combined active compliance, forces and torques remain small. In Figure 8a, the forces are reduced by a factor 9 (note the change in the coordinate scale). The effects of the tool changer clamping are considerably reduced. Similar improvements can be observed in torque behavior. In Figure 8b, the torques are




(a) (b)

Fig. 7. (a) Force and (b) torque evolution during peg-in-hole insertion with pure visual servo control ($Z$ = insertion axis, $X$ and $Y$ perpendicular plane).




(a) (b)

Fig. 8. (a) Force and (b) torque evolution during peg-in-hole insertion with combined visual and force control ($Z$ = insertion axis, $X$ and $Y$ perpendicular plane).

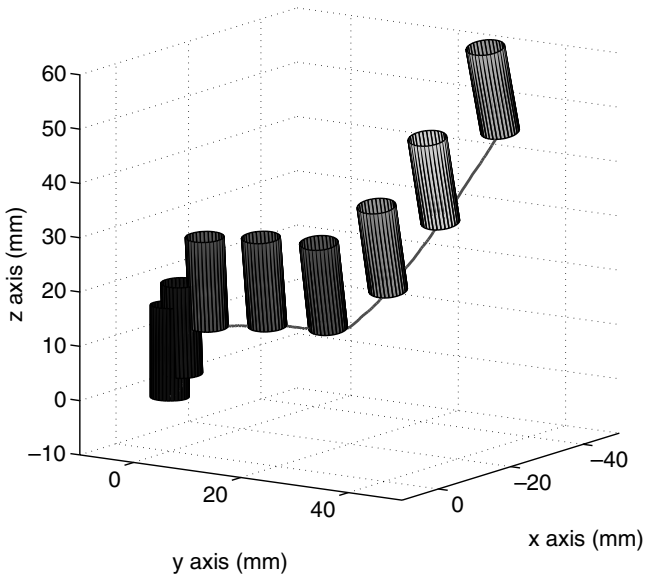



Fig. 9. Peg trajectory reconstructed from experimental data.

initially reduced by a factor 2. At $t = 13$ seconds, the tool changer is actuated and a sudden step appears on the plot. In contrast to the previous experiment, the force control allows us to decrease the torque by a factor of 9. This experimental result shows that the force feedback can compensate for the forces undesirably generated by two-dimensional visual feedback. Additional extensive experiments indicated that the force and vision combination is not limited to the limitation of undesirable forces. In Figure 9, the insertion experiment was realized with a very large initial positioning error in both position and orientation. It is clear that because the hole location is completely unknown, force control alone is not capable of performing this task. Figure 9 shows that our control scheme is capable of performing low-clearance peg-in-hole tasks with significant initial errors in all six degrees of freedom and without any knowledge of the hole location or the constraint geometry. No trajectory computation or complex insertion strategy is required, since the only input is the final desired image feature vector and the desired force.

## 6. Conclusions

The control approach proposed in this paper was designed to take into account several sensors to drive the end effector of a manipulator arm. The particular choice of the task function simplifies the design of the control law and the stability analysis. The stability analysis of the whole system is ensured by the stability of each subsystem. There are obvious advantages in combining sensor feedback in the control of a robot manipulator. With regard to vision sensors, previous work was based on stereo-camera configurations. Our paper extends the visual control to several cameras observing different objects or different parts of the same object. The experimental results obtained using two in-hand cameras show that it is possible to improve the positioning accuracy with respect to the use of a single camera. With regard to vision and force sensors, previous work was based on hybrid position/force control, which does not entirely exploit the usefulness of sensor duality. In this paper, we combined vision and force control within the impedance control approach. The implemented control scheme involved a pure damping position-based impedance control and an external image-based visual controller. It is simple and practical, since the impedance controller and the vision-based controller can be designed separately, as shown by stability analysis.

## Acknowledgments

This work was supported by Electricité de France (EDF). We are grateful to the team manager and the researchers of Direction des Etudes et Recherches at EDF for their participation and help.

## References

Agapakis, J. E. 1990. Approaches for recognition and interpretation of workpiece surface using structured lighting. *International Journal of Robotics Research* 9(5):3–16.

Allen, P., Timcenko, A., Yoshimi, B., and Michelman, P. 1993. Hand-eye coordination for robotic tracking and grasping. In *World Scientific Series in Robotics and Automated Systems*, Vol. 7, *Visual Servoing*, ed. K. Hashimoto, 33–69. Singapore: World Scientific Press.

Caccavale, F., Natale, C., Siciliano, B., and Villani, L. 1999. 6-dof impedance control based on angle/axis representation. *IEEE Transactions on Robotics and Automation* 15:289–300.

Cipolla, R., and Hollinghurst, N. 1997. Visually guided grasping in unstructured environment. *Robotics and Autonomous Systems* 19:337–346.

Eppinger, S. D., and Seering, W. P. 1989. Three dynamic problems in robot force control. *IEEE International Conference on Robotics and Automation*, Scottsdale, AZ, pp. 392–397.

Espiau, B. 1993. Effect of camera calibration errors on visual servoing in robotics. *3rd International Symposium on Experimental Robotics*, Kyoto, Japan.

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

Espiau, B., Merlet, J. P., and Samson, C. 1990. Force-feedback control and non-contact sensing: A unified approach. *Proceedings of the 8th CISM-IFTOMM Symposium on Theory and Practice of Robots and Manipulators.*

Hager, G. 1997. A modular system for robust positioning using feedback from stereo vision. *IEEE Transactions on Robotics and Automation* 13:582–595.

Heinrichs, B., Sepehri, N., and Thornton-Trump, A. 1996. Position-based impedance control of an industrial hydraulic manipulator. *IEEE International Conference on Robotics and Automation*, Vol. 1, Minneapolis, MN, pp. 284–290.

Hosoda, K., Igarashi, K., and Asada, M. 1996. Adaptive hybrid visual servoing/force control in unknown environment. *IEEE International Conference on Intelligent Robots and Systems*, Osaka, Japan, pp. 1097–1103.

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

Khatib, O. 1987. A unified approach for motion and force control: The operational space formulation. *IEEE Transactions on Robotics and Automation* 3(3):43–53.

Lawrence, D. 1988. Impedance control stability properties in common implementations. *IEEE International Conference on Robotics and Automation*, Philadelphia, pp. 1185–1192.

Malis, E. 1998. Contributions à la modélisation et à la commande en asservissement visuel. Ph.D. thesis, Université de Rennes I, Institut de Recherche en Informatique et Systèmes Aléatoires.

Malis, E., Chaumette, F., and Boudet, S. 1999. 2 1/2 d visual servoing. *IEEE Transactions on Robotics and Automation* 15:234–246.

Malis, E., Chaumette, F., and Boudet, S. 2000. Multi-cameras visual servoing. *IEEE International Conference on Robotics and Automation*, Vol. 4, San Francisco, pp. 3183–3188.

Malis, E., Morin, L., and Boudet, S. 1996. Control of redundant robots at singularities in degenerate directions. *12th CISM-IFToMM Symposium on Theory and Practice of Robots and Manipulators*, pp. 319–326.

Maru, N., Kase, H., Yamada, S., Nishikawa, A., and Miyazaki, F. 1993. Manipulator control by visual servoing with the stereo vision. *IEEE International Conference on Intelligent Robots and Systems*, Vol. 3, Yokohama, Japan, pp. 1866–1870.

Morel, G., Malis, E., and Boudet, S. 1998. Impedance based combination of visual and force control. *IEEE International Conference on Robotics and Automation*, Vol. 2, Leuven, Belgium, pp. 1743–1748.

Morel, G., Szewczyk, J., Boudet, S., and Pot, J. 1999. Explicit incorporation of 2d constraints in vision based control of robot manipulators. *Proceedings of ISER'99: Experimental Robotics IV*, Sydney, Australia, pp. 99–108.

Nelson, B. J., and Khosla, P. K. 1996. Force and vision resolvability for assimilating disparate sensory feedback. *IEEE Transactions on Robotics and Automation* 12:714–731.

Nelson, B. J., Morrow, J. D., and Khosla, P. K. 1996. Robotic manipulation using high bandwidth force and vision feedback. *Mathematical and Computational Modelling* 24(5-6):11–29.

Raibert, M. H., and Craig, J. J. 1981. Hybrid position/force control of manipulators. *ASME Journal of Dynamic Systems, Measurement and Control* 102:126–133.

Ruf, A., and Horaud, R. 1999. Rigid and articulated motion seen with an uncalibrated stereo rig. *IEEE International Conference on Computer Vision*, Corfu, Greece.

Samson, C., Le Borgne, M., and Espiau, B. 1991. *Oxford Engineering Science Series*, Vol. 2, *Robot Control: The Task Function Approach.* Oxford: Clarendon Press.

Whitney, D. E. 1977. Force feedback control of manipulator fine motions. *Journal of Dynamic Systems, Measurement and Control* 99(2):91–97.

Wilson, W. J., Hulls, C.C.W., and Bell, G. S. 1996. Relative end-effector control using Cartesian position-based visual servoing. *IEEE Transactions on Robotics and Automation* 12:684–696.