Multi-cameras Visual Servoing to Perform a Coordinated Task using a Dual Arm Robot

Renliw Fleurmond[1,2] and Viviane Cadenat[1,2]
[1] *CNRS, LAAS, 7 avenue du Colonel Roche, F-31400 Toulouse, France*
[2] *Univ. de Toulouse, UPS, LAAS, F-31400 Toulouse, France*

Keywords: Dual Arm Robots, Coordination, Vision-based control, Multi-cameras visual servoing.

Abstract: This paper deals with the problem of coordinating a dual arm robot equipped with several cameras. Our goal is to propose a vision-based control strategy allowing to realize a real cooperation of the two arms. The idea is to sequence different vision based tasks built from visual features describing the relative pose between the cap and the pen. Simulation results validate our approach.

1 INTRODUCTION

Dual arm manipulation has been studied since the eighties (Caccavale and Uchiyama, 2008). At the beginning, the objective was to enhance the range of realizable industrial applications. It then became possible to carry heavy loads (Bonitz and Hsia, 1996), manipulate flexible objects (Zheng and Chen, 1993), or assemble pieces (Yamada et al., 1995). Nevertheless, the recent development of service robots has led to the design of new mobile robotic systems able to help people in their daily life (co-worker, etc.) (Smith et al., 2012). To complete such missions, these robots must be able to perform complex manipulation tasks, which requires to coordinate both arms motion. According to (Zollner et al., 2004), coordination tasks can be divided into two classes : symmetric coordination when both arms manipulate the same object and asymmetric coordination where they carry different objects.

The coordination problem can be tackled at different levels and through many approaches (Smith et al., 2012). In particular, it has been shown that a pure position feedback is unable to deal with this kind of problem because of its sensitivity to modeling errors (Caccavale et al., 2001). Consequently, other approaches using exteroceptive data have been developed. Common solutions rely on hybrid position/force based control laws (Kraus and McCarragher, 1997; Watanabe et al., 2005) or active compliance control laws (Bonitz and Hsia, 1996; Albrichsfeld and Tolle, 2002). However, other kinds of exteroceptive data such as vision can be exploited, the robot being then controlled using visual servoing. This kind of control offers two main advantages which appear to be quite interesting in the context of dual arm manipulation. First, it can be realized using several (embedded and/or external) cameras (Kermorgant and Chaumette, 2011), which allows to benefit from complementary information on the task execution. Second, visual servoing, especially image based control, is known to offer nice robustness properties, which allows to accurately perform the task (Chaumette and Hutchinson, 2006). However, to the best of our knowledge, there are only few works which have really dealt with dual arm visual servoing (Smith et al., 2012). We present hereafter a brief overview. In (Miyabe et al., 2003) 2D visual servoing is used to simultaneously and independently control two arms to capture an object. In (Hynes et al., 2006) the two arms are alternatively controlled using visual servoing to tie surgical knots. Finally, Vahrenkamp et al. propose a 3D visual servoing to control the arms of a humanoid robot in order to grasp the handles of a wok or pour liquid from a bottle into a cup (Vahrenkamp et al., 2009). Thus, in these works, the coordination problem is not completely solved, the two arms being often separately or alternatively controlled.

In this paper, we address the coordination problem from a control point of view. We aim at designing a vision-based control strategy allowing to truly coordinate the motions of a dual arm robotic system equipped with several cameras. In other words, the control will be defined so that both arms *simultaneously* move to perform the desired task. Here, we

Fleurmond R. and Cadenat V..
Multi-cameras Visual Servoing to Perform a Coordinated Task using a Dual Arm Robot.
DOI: 10.5220/0005017700370044
In *Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics* (ICINCO-2014), pages 37-44
ISBN: 978-989-758-040-6

have chosen to realize an asymmetric coordination task consisting in recapping a pen. Our idea is to build an image based visual servoing (IBVS) control to benefit from its good robustness properties with respect to modeling errors (Chaumette and Hutchinson, 2006). This control law will be fed using the data provided by two different cameras (a fixed one and a mobile one) mounted on the robot. We will then develop a multi-camera visual servoing. To do so, as in (Uchiyama and Dauchez, 1988; Dauchez et al., 2005), we have considered the two arms as a single robotic system to be controlled. Furthermore, (Dauchez et al., 2005) and (Adorno et al., 2010) have shown that the realization of a coordination task requires to define it in terms of the relative pose between the two end-effectors. Here, we have followed a similar reasoning, except that our task will be directly expressed in the image plane instead of the 3D space, which will lead to the design of a more robust control law. This latter will be obtained by regulating to zero a relative error between the visual features describing the cap and the pen provided by the cameras.

This paper is organized as follows: the two next sections describe our contribution, namely the problem modeling and the proposed control strategy. Simulation results validating our approach are then presented while the last section is devoted to a conclusion and some prospects.

2 MODELING THE PROBLEM

2.1 Robot Model

Our robotic platform is the PR2 developed by Willow Garage. It consists of an omnidirectional mobile base equipped with two 7-DOF robotic arms. We consider that the mobile base and the head are fixed in this work. The robot has several cameras, and as explained before we will use the information provided by two of them to perform the task. The first one, whose center is denoted by C_f is fixed on the head and has a large field of view. The second one, whose center is denoted by C_m, is located on the right forearm. It gives us additional information on the scene which will be useful when assembling the two objects.

We introduce the different frames which will be necessary to model our problem (see figure 1). F_w (O_w, x_w, y_w, z_w) is fixed and represents the world frame. F_l (O_l, x_l, y_l, z_l) and F_r (O_r, x_r, y_r, z_r) are respectively the frames linked to the left and right end effectors. F_f (C_f, x_f, y_f, z_f) and F_m (C_m, x_m, y_m, z_m) are respectively the frames attached to fixed and mobile cameras. We denote by q_r and q_l (respectively $\dot{q}_r$ and $\dot{q}_l$) the joint coordinates (respectively the joint velocities) of the right and left arms.

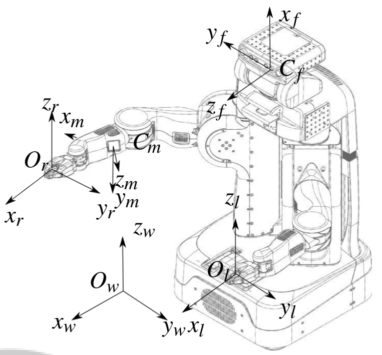

Figure 1: The robotic platform and the different frames.

Using these notations, the kinematic screws of the end effectors T_r and T_l, and of the mobile camera T_m with respect to F_w are given by the direct differential kinematic model as follows:

$$T_r = J_r \dot{q}_r, \quad T_l = J_l \dot{q}_l, \quad T_m = J_{5r} \dot{q}_r \tag{1}$$

where J_r, J_l and J_{5r} are 6×7 matrices which have been already determined. These kinematic screws are respectively expressed in frames F_r, F_l and F_m.

2.2 Modeling the Task

As previously mentioned, the task which is considered here consists in recapping a pen. To do so, we first introduce the two following hypotheses:

- **Hyp 1:** The cap and the pen are respectively gripped by the right and left arms.
- **Hyp 2:** The two objects are modeled by two cylinders as shown in figure 2. Furthermore, in the sequel, to avoid some singularities, we assume that these objects are always seen as shown in this figure.

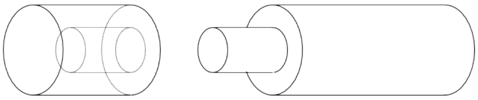

Figure 2: Cap and pen model.

To recap the pen, the robot has to align both cylinders before connecting them. But this movement cannot be done directly. We then divide the task into the three following subtasks as illustrated in figure 3 [1]:

- Make both cylinders axes *coplanar* while ensuring a sufficient distance between the two objects.

[1] The pen is shown as if it is fixed for the sake of clarity. But of course both objects are allowed to move.

- Make both cylinders axes *collinear* while ensuring a sufficient distance between the two objects.
- Maintain the alignment and *bring* the cap near the pen.

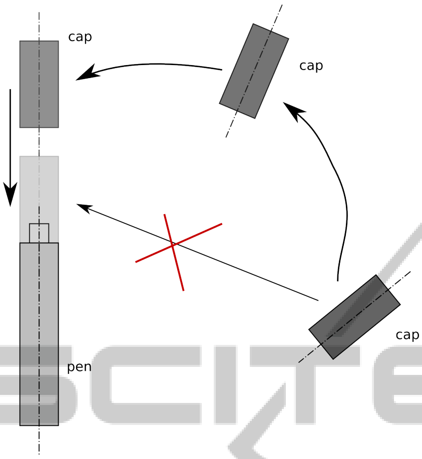

Figure 3: Steps to recap the pen.

2.3 Visual Features

In this part, our goal is to choose visual features allowing to represent the cap and the pen. Espiau et al. (Espiau et al., 1992) have used two lines to define the contour of the cylinder projection. Here we have chosen to consider the projection of the cylinder axis on the image plane. This straight line is obtained from the axis of the grey area (cf. figure 4) corresponding to the cylinder [2] in the image.

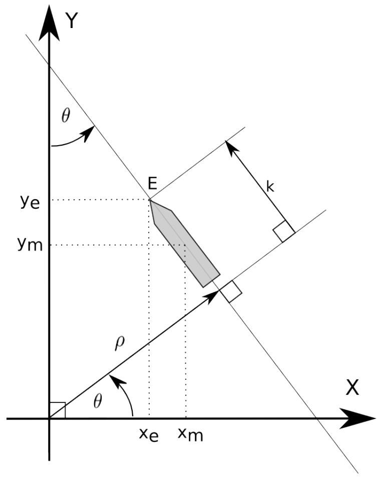

Figure 4: Visual features used: ρ, θ and *k*.

We have then represented the line by using polar parameters (ρ, θ) as in (Espiau et al., 1992). In this representation, ρ is the distance between the line and the center of the image, and θ is its relative orientation with respect to the vertical axis (cf. figure 4). To compute these features we have used the formulas given in (Berry et al., 2000). We get:

$$\begin{aligned}\theta &= \frac{1}{2}\,atan\left(\frac{2I_{xy}}{I_x - I_y}\right)\\ \rho &= \sqrt{x_m^2 + y_m^2}\end{aligned}$$

where (x_m, y_m) is the center of mass of the area represented in grey in the figure 4.

However, with such a representation, two lines which are symmetrical with respect to the origin are characterized using the same parameters. To overcome this ambiguity, we have allowed ρ to be positive or negative as in (Espiau et al., 1992). We have chosen to compute ρ by using this following expression:

$$\rho = x_m.\cos(\theta) + y_m.\sin(\theta) \tag{2}$$

I_{xy}, I_x and I_y are the second order moments of this area, and their well-known expressions are given by (Chaumette, 2002a):

$$\begin{aligned}A &= \sum_{area} 1\\ x_m &= \frac{1}{A}\sum_{area} x\\ y_m &= \frac{1}{A}\sum_{area} y\\ I_x &= \sum_{area}(x - x_m)^2\\ I_y &= \sum_{area}(y - y_m)^2\\ I_{xy} &= \sum_{area}(x - x_m)(y - y_m)\end{aligned}$$

Parameters θ and ρ allow us to partially control the orientation and the position of the objects. However, it is also necessary to monitor the translation of a cylinder along its axis. To this aim, we introduce a third parameter *k* which expresses the position of one end point E (x_e, y_e) of the cylinder on the straight line (cf. figure 4). We get:

$$k = y_e \cdot \cos(\theta) - x_e \cdot \sin(\theta) \tag{3}$$

Thus, the image of each cylinder is described by three parameters. As we consider two cylindrical objects and two cameras, we define the four following visual features vectors:

[2] It is not an exact value of the projection of the axis but it provides a good approximation.

$$\begin{array}{cc}
\multicolumn{2}{c}{\textbf{Cap}} \\
\text{Fixed camera} & \text{Moving camera} \\
S_{fc} = [\rho_{fc}, \theta_{fc}, k_{fc}]^T & S_{mc} = [\rho_{mc}, \theta_{mc}, k_{mc}]^T \\
\multicolumn{2}{c}{\textbf{Pen}} \\
\text{Fixed camera} & \text{Moving camera} \\
S_{fp} = [\rho_{fp}, \theta_{fp}, k_{fp}]^T & S_{mp} = [\rho_{mp}, \theta_{mp}, k_{mp}]^T
\end{array}$$

Remark: A singularity occurs in the representation of the visual features if the area is a disc because $I_x = I_y$ and $I_{xy} = 0$. However this case cannot happen here thanks to hypothesis 2.

3 CONTROL STRATEGY

To perform our task, we have chosen to use visual servoing which allows to control a robot using the visual informations provided by one or several cameras (Chaumette and Hutchinson, 2006). Visual servoing can be roughly speaking divided into three classes: 3D, 2D, 2D-1/2 (Chaumette and Hutchinson, 2006). Here we have used the second approach also known as image based visual servoing (IBVS), because of its well-known properties of robustness with respect to errors (Chaumette and Hutchinson, 2006).

To design our control law we have used the task function approach (Samson et al., 1991) where the task is described by a n-dimensional C^2 function $e(q,t)$, to be regulated to zero.

As classically done in the visual servoing area (see for example (Espiau et al., 1992; Chaumette and Hutchinson, 2006)), we impose an exponential decay to make e(q,t) vanish. This controller is given by (Espiau et al., 1992) :

$$\dot{q} = -J^{+}\lambda e \tag{4}$$

where λ is a positive gain or a positive-definite matrix, J the jacobian of the task function and J^+ its Moore-Penrose inverse.

Our goal is to use this formalism to design our control law. To this aim, we first consider the two arms as a single robotic system as in (Uchiyama and Dauchez, 1988; Dauchez et al., 2005). The control vector will then be composed of the joint velocities of both arms: $\dot{q} = \begin{bmatrix} \dot{q}_r^T & \dot{q}_l^T \end{bmatrix}^T$. Moreover, it is necessary to define the task functions corresponding to the above mentioned three tasks. In 2D visual servoing $e(q,t)$ is generally given by an error between the current vector of visual features S and the desired ones S^*, S^* being often constant. However, if we follow this reasoning to define our task functions, we will monitor the absolute pose of the two objects with respect to the world frame, and the arms will be separately controlled. Therefore, to really coordinate the two arms, it is truly better to focus on the relative pose between the end effectors, as proposed in (Adorno et al., 2010; Dauchez et al., 2005). Following the same idea, we express the task functions as a relative error between the visual features representing to the cap and the pen.

3.1 The Task Functions

In this part, the aim is to model the previously mentionned subtasks by three task functions e_1, e_2, e_3. We consider the first one. To perform this substask, it is necessary to constrain the two cylinder axes to belong to the same plane. To do so, it suffices to align their projection in one image. We have chosen to use the one provided by the mobile camera. The first subtask is then expressed as follows:

$$e_1 = \begin{bmatrix} \rho_{mc} - \rho_{mp} \\ \theta_{mc} - \theta_{mp} \\ k_{mc} - k_{mp} - d \end{bmatrix} \tag{5}$$

where d is a constant value allowing to maintain a given distance between the cap and the pen.

Now, we consider the second subtask. Its goal is to align the axes of the two cylinders. At least two points of view are necessary to guarantee that two straight lines are aligned. Therefore we have to consider the visual features provided by both cameras, which leads to implement a multi-cameras visual servoing control law. Our second subtask is defined as shown below:

$$e_2 = \begin{bmatrix} e_1 \\ \rho_{fc} - \rho_{fp} \\ \theta_{fc} - \theta_{fp} \end{bmatrix} \tag{6}$$

Finally, our last subtask has to keep the axes aligned while bringing the cap to the pen. We propose the following expression:

$$e_3 = \begin{bmatrix} S_{mc} - S_{mp} \\ S_{fc} - S_{fp} \end{bmatrix} \tag{7}$$

Taking into account the expressions of e_1, e_2, and e_3, it is possible to show that these subtasks can be re-written as follows:

$$e_i = H_i \cdot \begin{bmatrix} S_{mc} - S_{mp} - A_i \\ S_{fc} - S_{fp} \end{bmatrix} \tag{8}$$

where H_i is an activation matrix which allows to select only the necessary visual features. H_i and A_i are defined for each subtask e_i as shown below:

$$\begin{array}{ll}
H_1 = \begin{bmatrix} I_{3\times3} & 0_{3\times3} \end{bmatrix} & \text{and } A_1 = [0,0,d]^T \\
H_2 = \begin{bmatrix} I_{5\times5} & 0_{5\times1} \end{bmatrix} & \text{and } A_2 = [0,0,d]^T \\
H_3 = I_{6\times6} & \text{and } A_3 = 0_{3\times1}
\end{array}$$

Once the task functions are defined, we now focus on the control law allowing to regulate them to zero.

3.2 Control Design

To design the control law, our idea is to use equation (4). To this aim, we have to determine the jacobian of each task function. Using the general formulation (8), the time derivative of our subtasks e_i is given by:

$$\dot{e}_i = J_i\dot{q},\ \dot{e}_i = H_i \cdot \begin{bmatrix} \dot{S}_{mc} - \dot{S}_{mp} \\ \dot{S}_{fc} - \dot{S}_{fp} \end{bmatrix} \tag{9}$$

Following (Chaumette, 2002b), the time derivative $\dot{S}$ of a given visual features vector denoted by S expresses as follows:

$$\dot{S} = L \cdot (T_c - T_o) \tag{10}$$

where L is the interaction matrix, T_c and T_o are respectively the kinematic screws of the camera and of the mobile target with respect to the world frame F_w. They are expressed in the camera frame. Let us recall that we consider two cameras: a mobile one and a static one. We have to express T_c and T_o in both cases.

3.2.1 Fixed Camera

In this case, the camera kinematic screw T_c is zero. Thus it remains to compute T_o. Here T_o represents the kinematic screw of the moving object (cap or pen). This screw must be expressed at a particular point of the object which coincides at every moment with the fixed camera center C_f. To determine it, let us recall the following well-known relation (Pérez, 1989):

$$T_b = {}^bM_aT_a \text{ where } {}^bM_a = \begin{bmatrix} I_{3\times3} & [AB]_\times \\ 0 & I_{3\times3} \end{bmatrix}$$

where T_a and T_b are the kinematic screws of two points A and B belonging to the same mobile solid. They are expressed with respect to a fixed frame. $AB_\times$ is the skew-symmetric matrix such as $\forall\vec{V} \in \mathbb{R}^3$, $[AB]_\times\vec{V} = \overrightarrow{AB} \times \vec{V}$.

Recalling that the cap is moved by the left end effector and the pen by the right one, and that T_o must be expressed in the camera frame F_f, the following expressions yield:

- For the cap

$$T_o = \begin{bmatrix} {}^fR_r & 0_{3\times3} \\ 0_{3\times3} & {}^fR_r \end{bmatrix} {}^fM_rT_r = {}^fW_rT_r$$

- For the pen

$$T_o = \begin{bmatrix} {}^fR_l & 0_{3\times3} \\ 0_{3\times3} & {}^fR_l \end{bmatrix} {}^fM_lT_l = {}^fW_lT_l$$

where ${}^fR_r$ and ${}^fR_l$ are respectively the rotation matrices between F_r and F_f, and F_l and F_f.

Now combining these expressions and the equations (1) and (10), the time derivatives of S_{fc} and S_{fp} are:

$$\begin{aligned} \dot{S}_{fc} &= -L_{fc}{}^fW_rJ_r\dot{q}_r && (11) \\ \dot{S}_{fp} &= -L_{fp}{}^fW_lJ_l\dot{q}_l && (12) \end{aligned}$$

where L_{fc} and L_{fp} are the interaction matrices corresponding to S_{fc} and S_{fp}. Their expressions are detailed in appendix A.

3.2.2 Moving Camera

In this case, the camera and the two objects are both moving. We have then to determine the kinematic screws T_c and T_o. The first one is already known and is given by T_m (see equation (1)).

It then remains to compute T_o. We follow the same reasoning as previously, keeping in mind that the kinematic screw must be expressed at the point which coincides with the center C_m of the mobile camera. We finally obtain the following result:

- For the cap

$$T_o = \begin{bmatrix} {}^mR_r & 0_{3\times3} \\ 0_{3\times3} & {}^mR_r \end{bmatrix} {}^mM_rT_r = {}^mW_rT_r$$

- For the pen

$$T_o = \begin{bmatrix} {}^mR_l & 0_{3\times3} \\ 0_{3\times3} & {}^mR_l \end{bmatrix} {}^mM_lT_l = {}^mW_lT_l$$

where ${}^mR_r$ and ${}^mR_l$ are respectively the rotation matrices between F_r and F_m and F_l and F_m.

Using (10) and (1), we obtain the time derivative of S_{mc} and S_{mp}:

$$\begin{aligned} \dot{S}_{mc} &= L_{mc}(J_{5r}\dot{q}_r - {}^mW_rJ_r\dot{q}_r) && (13) \\ \dot{S}_{mp} &= L_{mp}(J_{5r}\dot{q}_r - {}^mW_lJ_l\dot{q}_l) && (14) \end{aligned}$$

where L_{mc} and L_{mp} are the interaction matrices corresponding to S_{mc} and S_{mp}. Their expressions are detailed in appendix A.

3.2.3 The Jacobian Matrix of Each Subtask

Finally we can deduce that the jacobian of each subtask e_i is given by:

$$J_i = H_i \cdot \begin{bmatrix} -L_{mc}{}^mW_r & L_{mp}{}^mW_l & L_{mc} - L_{mp} \\ -L_{fc}{}^fW_r & L_{fp}{}^fW_l & 0_{3\times6} \end{bmatrix} \cdot \begin{bmatrix} J_r & 0_{6\times7} \\ 0_{6\times7} & J_l \\ J_{5r} & 0_{6\times7} \end{bmatrix} \tag{15}$$

The control law corresponding to each subtask e_i is then expressed as follows:

$$\dot{q}_i = -J_i{}^+\lambda_ie_i \tag{16}$$

with $i = \{1,2,3\}$.

3.3 Transition Between Tasks

We have defined three task functions. To recap the pen, it is necessary to sequence them. In other words, we need to sequence the control laws allowing to regulate them to zero, while preserving the smoothness of the velocities applied to the robot.

This objective cannot be performed by considering the classical exponential decrease which is generally set to make the task function vanish. A first solution to this problem has been proposed in (Soueres et al., 2003). The idea is to impose a second order linear dynamic to make the task function vanish.

$$\ddot{e} + \alpha\dot{e} + \beta e = 0$$

Thanks to this choice, it is possible to define the initial conditions on e and $\dot{e}$ so that the control law smoothness is guaranteed. A year later, (Mansard and Chaumette, 2004) have proposed a particular choice for alpha and beta, leading to the following linear differential equation:

$$\ddot{e} + (\lambda + \mu)\dot{e} + \lambda\mu e = 0$$

The main advantage of this solution is that it allows to separately choose λ and μ. If $\mu > \lambda > 0$, the system is stable and the transient time is set only by μ. So, to switch from a task e_{i-1} to another task e_i at t_i, the real control $\dot{q}$ sent to the robot is given by the following expression:

$$\dot{q}(t) = \dot{q}_i(t) - \exp(-\mu(t - t_i))(\dot{q}_i(t_i) - \dot{q}(t_i)) \quad (17)$$

where $\dot{q}_i$ is the value computed by the formula (16) and t_i the switching time.

We switch from the current control law to the next one when the norm of the current task function drops below a given threshold.

4 SIMULATION RESULTS

We now present simulation results to validate our approach. A gaussian noise has been introduced on the visual features (mean = 0, standard deviation = 2 pixels) [3], on the joint coordinates q (mean = 0, covariance = 7.6×10^{-7} rad^2) and on the velocities sent to the robot (mean = 0 rad/s, covariance = 7.6×10^{-7} rad^2/s^2). The control law is updated at 15 Hertz rate. Both arms trajectory has been recorded and played [4] using ROS [5] on the Gazebo simulator [6].

The evolution of the components T_i of the three task functions e_1, e_2 and e_3 is presented on figure 5. As we can see, all of them converge towards zero, which shows that the corresponding tasks are successfully achieved. The first subtask e_1 is active between 0 and 2.4 s: in this interval T_1, T_2 and T_3 are respectively defined by $\rho_{mc} - \rho_{mp}$, $\theta_{mc} - \theta_{mp}$, and $k_{mc} - k_{mp} - d$.

At instant $2.4s$, the norm of e_1 drops under a threshold equal to 10^{-2}, we start realizing the second subtask. At this time, T_4 and T_5 are given by $\rho_{fc} - \rho_{fp}$ and $\theta_{fc} - \theta_{fp}$. At $t = 4.35s$, the value of e_2 becomes smaller than the chosen threshold and the last subtask is launched. T_3, T_4 and T_5 are then respectively defined by $\rho_{fc} - \rho_{fp}$, $\theta_{fc} - \theta_{fp}$ and $k_{mc} - k_{mp}$. The two switching instants are shown by two black lines on figure 5.

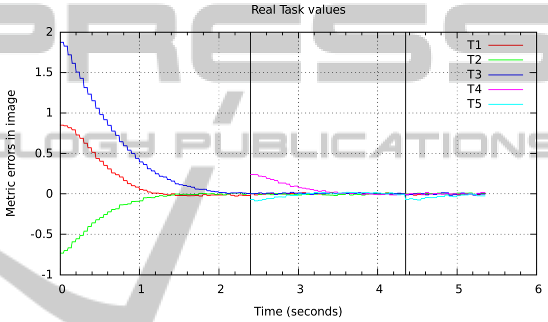

Figure 5: Evolution of the task function.

Figures 6 and 7 show the evolution of the velocities sent to the two arms. As we can see, the control inputs applied to each of them appear to be similar, which shows that both arms move to achieve the task and that the coordination between them is properly performed. These figures also demonstrate the efficiency of our smoothness strategy, as no sudden jumps is registered.

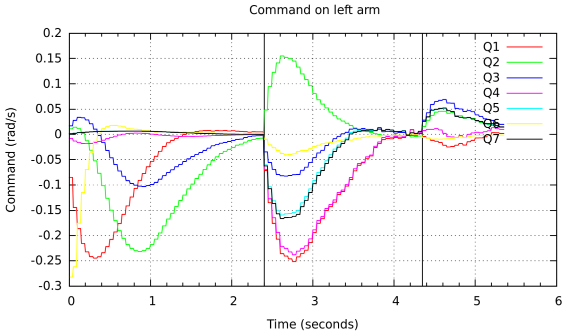

Figure 6: Control sent to the right arm.

[3] The sensor has the size of $4.51mm \times 2.88mm$ with a focal length of $2.5mm$ and provides an image of 752×480 pixels.

[4] A video is available on http://homepages.laas.fr/rfleurmo/.

[5] http://www.ros.org/

[6] http://gazebosim.org/

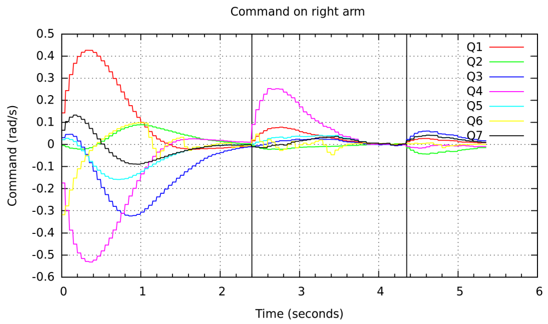

Figure 7: Control sent to the left arm.

5 CONCLUSION

In this paper, we have tackled the problem of coordinating two manipulator arms from a control point of view. We have proposed a vision-based control strategy allowing to truly coordinate the motions of a dual-arm robotic system. The task to be performed has been described by a sequence of three subtasks. Each of them is defined by visual features characterizing the relative pose between the end effectors so that a true collaboration between both arms has been achieved. A multi-cameras image based visual servoing has then been designed. Finally, the proposed control strategy has been validated and the obtained simulation results have demonstrated its interest and its efficiency.

Now, to go further and improve our approach, it is necessary to take into account the unexpected events which may occur during the task and hamper its execution (e.g., joint limits, singularities, collisions, occlusions, etc.). Therefore, our next step will be to address this problem by taking advantage of the system redundancy. In addition to these theoretical improvements, we also plan to experimentally validate our approach on the LAAS PR2 robot and to perform more complex coordination tasks.

REFERENCES

Adorno, B., Fraisse, P., and Druon, S. (2010). Dual position control strategies using the cooperative dual task-space framework. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 3955–3960.

Albrichsfeld, C. V. and Tolle, H. (2002). A self-adjusting active compliance controller for multiple robots handling an object. *Control Engineering Practice*, 10(2):165 – 173.

Berry, F., Martinet, P., and Gallice, J. (2000). Turning around an unknown object using visual servoing. In *Intelligent Robots and Systems, 2000. (IROS 2000). Proceedings. 2000 IEEE/RSJ International Conference on*, volume 1, pages 257–262 vol.1.

Bonitz, R. and Hsia, T. (1996). Robust internal-force based impedance control for coordinating manipulators-theory and experiments. In *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, volume 1, pages 622–628 vol.1.

Caccavale, F., Ciro, N., Siciliano, B., and Villani, L. (2001). Achieving a cooperative behavior in a dual-arm robot system via a modular control structure. *Journal of Robotic Systems*, 18(12):691–699.

Caccavale, F. and Uchiyama, M. (2008). Cooperative manipulators. In Siciliano, B. and Khatib, O., editors, *Springer Handbook of Robotics*, pages 701–718. Springer Berlin Heidelberg.

Chaumette, F. (2002a). A first step toward visual servoing using image moments. In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, volume 1, pages 378–383 vol.1.

Chaumette, F. (2002b). *La commande des robots manipulateurs: Asservissement visuel*, chapter 3, pages 101 – 151.

Chaumette, F. and Hutchinson, S. (2006). Visual servo control part 1: Basic approaches. *IEEE Robotics and Automation Magazine*, 13(4):82–90.

Dauchez, P., Fraisse, P., and Pierrot, F. (2005). A vision/position/force control approach for performing assembly tasks with a humanoid robot. In *5th IEEE-RAS International Conference on Humanoid Robots*, pages 277–282.

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

Hynes, P., Dodds, G., and Wilkinson, A. J. (2006). Uncalibrated visual-servoing of a dual-arm robot for mis suturing. In *The First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics*, pages 420–425.

Kermorgant, O. and Chaumette, F. (2011). Multi-sensor data fusion in sensor-based control application to multi-camera visual servoing. In *IEEE International Conference on Robotics and Automation*, pages 4518–4523.

Kraus, W., J. and McCarragher, B. J. (1997). Hybrid position/force coordination for dual-arm manipulation of flexible materials. In *Intelligent Robots and Systems, 1997. IROS '97., Proceedings of the 1997 IEEE/RSJ International Conference on*, volume 1, pages 202–207 vol.1.

Mansard, N. and Chaumette, F. (2004). Tasks sequencing for visual servoing. In *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 1, pages 992–997 vol.1.

Miyabe, T., Konno, A., and Uchiyama, M. (2003). Automated object capturing with a two-arm flexible manipulator. In *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, volume 2, pages 2529–2534 vol.2.

Pérez, J. (1989). *Mécanique: points matériels, solides, fluides : avec exercices et problèmes résolus*. Enseignement de la physique. Masson.

Samson, C., Borgne, M. L., and Espiau, B. (1991). *Robot control: the task function approach*. Oxford engineering science series. Clarendon Press.

Smith, C., Karayiannidis, Y., Nalpantidis, L., Gratal, X., Qi, P., Dimarogonas, D., and Kragic, D. (2012). Dual arm manipulationa survey. *Robotics and Autonomous Systems*, 60(10):1340 – 1353.

Soueres, P., Cadenat, V., and Djeddou, M. (2003). Dynamical sequence of multi-sensor based tasks for mobile robots navigation. *7th Symposium on Robot Control (SYROCO'03)*, 2:423–428.

Uchiyama, M. and Dauchez, P. (1988). A symmetric hybrid position/force control scheme for the coordination of two robots. In *Robotics and Automation, 1988. Proceedings., 1988 IEEE International Conference on*, pages 350–356 vol.1.

Vahrenkamp, N., Boge, C., Welke, K., Asfour, T., Walter, J., and Dillmann, R. (2009). Visual servoing for dual arm motions on a humanoid robot. In *9th IEEE-RAS International Conference on Humanoid Robots*, pages 208–214.

Watanabe, T., Harada, K., Jiang, Z., and Yoshikawa, T. (2005). Object manipulation under hybrid active/passive closure. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 1013–1020.

Yamada, Y., Nagamatsu, S., and Sato, Y. (1995). Development of multi-arm robots for automobile assembly. In *Robotics and Automation, 1995. Proceedings., 1995 IEEE International Conference on*, volume 3, pages 2224–2229 vol.3.

Zheng, Y. and Chen, M. (1993). Trajectory planning for two manipulators to deform flexible beams. In *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, pages 1019–1024 vol.1.

Zollner, R., Asfour, T., and Dillmann, R. (2004). Programming by demonstration: Dual arm manipulation tasks for humanoids robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*.

APPENDIX A

We briefly present the computation of the interaction matrix corresponding to the above mentioned visual features. Using the well-known pinhole camera model, it is possible to express the projection of a point M on the image plane as follows:

$$x = fX/Z,\ y = fY/Z$$

where f is the focal length, (x,y) the coordinates of the projected point in the image and (X,Y,Z) the coordinates of M with respect to the camera frame. The expression of the interaction matrix of a point is given by:

$$L_{xy} = \begin{bmatrix} -f/Z & 0 & x/Z & xy/f & -(f+x^2/f) & y \\ 0 & -f/Z & y/Z & f+y^2/f & -xy/f & -x \end{bmatrix}$$

The interaction matrix $L_{\rho\theta}$ of a straight line described by the parameter vector $[\rho,\theta]^T$ has also already been computed in (Chaumette, 2002b):

$$L_{\rho\theta} = \begin{bmatrix} -1 & y_a \cdot cos(\theta) - x_a \cdot sin(\theta) \\ -1 & y_b \cdot cos(\theta) - x_b \cdot sin(\theta) \end{bmatrix}^{-1} \times$$

$$\begin{bmatrix} -cos(\theta) & -sin(\theta) & 0 & 0 \\ 0 & 0 & -cos(\theta) & -sin(\theta) \end{bmatrix} \times \begin{bmatrix} L_a \\ L_b \end{bmatrix}$$

where L_a and L_b are the interaction matrices corresponding to two distinct points $A(x_a,y_a)$, $B(x_b,y_b)$ which belong to the line.

Finally, the interaction matrix L_k relative to k can be expressed using the previous formulas:

$$L_k = \begin{bmatrix} -sin(\theta) & cos(\theta) & 0 & (-x_e \cdot cos(\theta) - y_e \cdot sin(\theta)) \end{bmatrix} \begin{bmatrix} L_e \\ L_{\rho\theta} \end{bmatrix}$$

where L_e is the interaction matrix corresponding to the point E.

The interaction matrix of the visual features vector, describing the cylinder is finally given by: $L_{cyl} = \begin{bmatrix} L_{\rho\theta}{}^T & L_k{}^T \end{bmatrix}^T$.