Multi-cameras visual servoing

E. Malis, François Chaumette, S. Boudet

Multi-cameras visual servoing

Ezio MALIS

IRISA/INRIA
Campus de Beaulieu
35042 Rennes, France

François CHAUMETTE

IRISA/INRIA
Campus de Beaulieu
35042 Rennes, France

Sylvie BOUDET

EDF-DER
6 Quai Watier
78401 Chatou, France

Abstract

In this paper, the classical visual servoing techniques have been extended to the use of several cameras observing different parts of an object. The multi-camera visual servoing has been designed as a part of the task function approach. The particular choice of the task function allows us to simplify the design of the control law and the stability analysis. A positioning task on a cumbersome object has been realized using 2D and $2\frac{1}{2}D$ visual servoings with two cameras, mounted on a manipulator robot, and observing two different parts of the object.

1 Introduction

Introducing several cameras in a visual servoing loop allow us to have richer information on the observed scene. Indeed, visual servoing systems can be classified as a function of the number of cameras used. The most part of these systems use only one camera since the hardware and software architecture is simplified, the image processing time is minimal, and the cost is reduced. However, the loss of the depth information can complicate the design of the scheme if the depth has to be estimated on-line. A solution to this problem is the utilisation of several cameras in a configuration such that the desired target is in the field of view of all the cameras (binocular, trinocular vision...).

The most used configuration using two cameras is called stereo (i.e. two cameras observing the same part of the scene) since it allows, if the system is calibrated, to obtain a complete Euclidean reconstruction of the observed objects. For example, in [1], a stereo pair is used to reconstruct the position of a moving objet at video rate. Then, the end-effector of the robot is controlled in order to track and pick the object. A stereo head can be used to perform 2D visual servoing like in [9], where the information given by two cameras mounted on a robot end-effector are used to update the Jacobian of the task. This system is "End-point Open Loop" and, as for mono-camera systems, the positioning precision depends on the calibration of the system if a teaching by showing technique cannot be used.

Even if the 3D model of the target is not needed to compute the displacement of the robot, the calibration of the stereo head is a problem as difficult as the calibration of a single camera. This is the reason why the "End-point Closed Loop" systems are more often used. It must be remembered that these systems need one or several out-hand cameras in order to observe at the same time the robot end-effector and the target. The positioning task can thus be carried out by controlling the error between the image of the end-effector and the image of the target. Such a system has been proposed in [5] to control a six d.o.f. robot using a 2D visual servoing technique where the considered visual information are projective invariants obtained from an uncalibrated stereo head. Another method in image-based visual servoing consists in numerically estimating the coefficients of the interaction matrix, without taking into account its analytical form [6]. This method does not need any 3D a priori knowledge and could be applied to multi-cameras visual servoing. However, it is unfortunately impossible to demonstrate under which conditions ensure its stability.

A 3D visual servoing technique, controlling a five d.o.f. robot and using a not-in-hand stereo head, was proposed in [2]. This system uses an affine camera model, instead of a projective model, in order to reconstruct, from the observed features, the position of some reference planes on the target and on the end-effector. Then, a 3D visual servoing technique is used to position the robot. It must be noticed that the linear approximation of the camera model simplifies the estimation and the control law of the robot, but it is valid only locally. Consequently, if the camera displacement is too big, this method may not converge. A last promising stereo technique, which does not need the cameras calibration nor the hand-eye calibration, was proposed in [10]. This method is based on the estimation of a (4×4) projective homography matrix which relates two sets of points of the projective space. The robot is then controlled without doing any Euclidean reconstruction.

Even if using additional cameras in visual servoing schemes brings a greater quantity of information, the target must stay in the field of view of all the cameras. The more cameras are used, the more it is difficult to have a big zone of intersection of their points of view. Furthermore, there exist some tasks where the fields of

view of the cameras cannot overlap, as for example the positioning with respect to a cumbersome object. In order to maintain the generality, the general configuration with two cameras will be called a bi-cameras configuration (the stereo configuration being a particular case of a bi-cameras configuration). Therefore, it is also interesting to study the case of a multi-cameras visual servoing not necessarily stereo. Thus, we would like to generalise the classical control laws to the use of several cameras whose fields of view do not intersect.

2   A multi-cameras task

A multi-cameras visual servoing task of maintenance in the steam generator of a nuclear power plant is now described. The task consists to close the entry of the primary circuit of a steam generator with a circular cover having a diameter of 1 meter (see Figure 1).
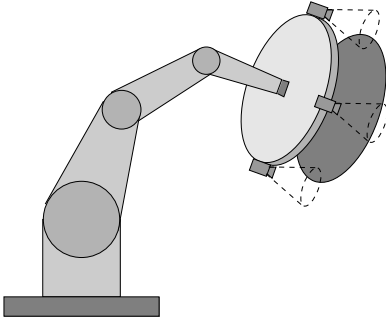

Figure 1: Cover positioning in a steam generator

The cameras are placed on the robot around the cover in order to observe the edges of the hole. Due to the big dimension of the cover, it is evident that:

- using only one camera could produce a "lever effect", i.e. a little error on the side near the camera could produce a bigger error on the other side of the cover. Using more cameras is thus desirable;

- the fields of view of the cameras cannot overlap, excepted if several stereo pairs are used, which is an unrealistic hypothesis.

In the Figure 1, the task was represented with three cameras but their number and their disposition around the cover is a problem still open. The following hypothesis is made: *little errors on the position of visual features observed by a camera imply that at least one other camera of the system is "active" (i.e. the target is in its field of view)*. It must be noticed that this hypothesis is independent on the calibration of the system, but it depends only on the precision which is possible to obtain from each camera and its placement in the system.

This hypothesis is then true if two cameras are "close enough" and it can help us to determine the minimal number of cameras necessary to realise the task.

3   Modelling

Let $\mathbf{s}_i$ be the n_i dimensional vector containing the sensor signals (i.e. feature points coordinates), and let $\mathbf{s} = \begin{bmatrix} \mathbf{s}_1^T & \mathbf{s}_2^T & \ldots & \mathbf{s}_N^T \end{bmatrix}^T$ be the n dimensional vector (where $n = \sum_{i=1}^N n_i$) containing all the signals coming from N sensors. Let us suppose that each sensor can control all the end-effector d.o.f. of the robot, then $n_i \geq m$ (where generally $m = 6$). Let us remark that, even if we are interested in a visual servoing task, the same scheme can be applied to different sensors.

Let be $\mathbf{r}$ the six dimensional vector containing the position and the orientation of the end-effector. The relationship between the time derivative of the visual information and the end-effector velocity is:

$$\dot{\mathbf{s}} = \begin{bmatrix} \mathbf{L}_1 & 0 & \ldots & 0 \\ 0 & \mathbf{L}_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \mathbf{L}_N \end{bmatrix} \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \\ \vdots \\ \mathbf{W}_N \end{bmatrix} \dot{\mathbf{r}} = \mathbf{L}\mathbf{W}\dot{\mathbf{r}} \quad (1)$$

where $\mathbf{L}_i$ is the interaction matrix relative to the ith camera, $\mathbf{W}_i$ is the transformation matrix between the velocity of the ith camera and the robot end-effector velocity, and $\mathbf{L}_i\mathbf{W}_i$ is the image Jacobian matrix. Each interaction matrix depends on the distance of the target to the camera. When an object is in the field of view of at least two cameras, the distance can be estimated. Otherwise, it can be approximated for each camera, exactly as for the mono-camera visual servoing [4], when the desired features in the images $\mathbf{s}^*$ are stored.

We use a task function of the form $\mathbf{e} = \mathbf{C}(\mathbf{s} - \mathbf{s}^*)$ [4], where $\mathbf{C}$ is a full rank constant matrix of dimensions $(m \times n)$, which allows to take into account the information redundancy. The time derivative of the task function is:

$$\dot{\mathbf{e}} = \mathbf{C}\dot{\mathbf{s}} = \mathbf{C}\mathbf{L}\mathbf{W}\dot{\mathbf{r}} \quad (2)$$

The matrix $\mathbf{C}$ must be chosen such that the matrix $\mathbf{C}\mathbf{L}\mathbf{W}$ has full rank. Different choices of $\mathbf{C}$ are possible:

- $\mathbf{C} = (\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^+$ is the pseudo-inverse of the Jacobian matrix calculated at the desired position $\widehat{\mathbf{L}}^* = \mathbf{L}(\mathbf{s}^*, \mathbf{g}^*)$. This choice ensure 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 realise the decoupling of the control law near the convergence.

- $\mathbf{C} = (\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^\sharp$ is the block pseudo-inverse of the Jacobian matrix:

$$(\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^\sharp = \left[\begin{array}{ccc} \kappa_1 \widehat{\mathbf{W}}_1^{-1}\widehat{\mathbf{L}}_1^{*+} & \ldots & \kappa_N \widehat{\mathbf{W}}_N^{-1}\widehat{\mathbf{L}}_N^{*+} \end{array} \right] \tag{3}$$

where $\widehat{\mathbf{L}}_i^{*+}$ is the pseudo-inverse of $\widehat{\mathbf{L}}_i^*$, $\widehat{\mathbf{W}}_i^{-1}$ is the inverse of $\widehat{\mathbf{W}}_i$ and κ_i is a positive weighting factor such that $\sum_{i=1}^{N}\kappa_i = 1$. This choice presents intermediate characteristics with respect to the two precedent choices.

It must be noticed that all the matrices $\mathbf{C}$ given above have rank m. Their null space is therefore of dimension $n - m$. Only the basis of the null space is different for each matrix. The interest of using one or the other choice will be shown when computing the control law and when analysing the stability of the system. However, it can be noticed now that the last choice presents some interesting characteristics. Indeed, if we consider for each camera a task function $\mathbf{e}_i = \mathbf{C}_i\mathbf{s}_i$ with $\mathbf{C}_i = \widehat{\mathbf{W}}_i^{-1}\widehat{\mathbf{L}}_i^+$, then the task function of the entire system is a weighted mean of the task function relative to each camera:

$$\mathbf{e} = \mathbf{C}\dot{\mathbf{s}} = \sum_{i=1}^{N}\kappa_i\mathbf{e}_i \tag{4}$$

Such a choice of the matrix $\mathbf{C}$ leads to a bigger modularity. A task function for each sub-system can be designed and the sensor fusion can be done at a higher level. For example, in the case of information loss due to the failure of a camera, it is sufficient to do not add the corresponding task function.

4 Control law

If a simple proportional control law is used by imposing the exponential decreasing of the task function $\dot{\mathbf{e}} = -\lambda\mathbf{e}$, the ideally decoupling control law is:

$$\dot{\mathbf{r}} = -\lambda(\mathbf{CLW})^{-1}\mathbf{e} \tag{5}$$

A more realistic control law is the following:

$$\dot{\mathbf{r}} = -\lambda(\mathbf{C}\widehat{\mathbf{L}}\widehat{\mathbf{W}})^{-1}\widehat{\mathbf{e}} \tag{6}$$

where the hats mean that the parameters are estimated or approximated. We choose, as in [4], to use in the control law the interaction matrix calculated at the desired position: $\widehat{\mathbf{L}} = \mathbf{L}^*(\mathbf{s}^*, \widehat{\mathbf{g}}^*)$. Consequently, choosing $\mathbf{C} = (\widehat{\mathbf{L}}\widehat{\mathbf{W}})^\sharp$ defined by equation (3), we get:

$$(\mathbf{C}\widehat{\mathbf{L}}\widehat{\mathbf{W}})^{-1} = (\sum_{i=1}^{N}\kappa_i\widehat{\mathbf{W}}_i^{-1}\widehat{\mathbf{L}}_i^{*+}\widehat{\mathbf{L}}_i^*\widehat{\mathbf{W}}_i)^+ = \sum_{i=1}^{N}\kappa_i\mathbf{I}_m = \mathbf{I}_m \tag{7}$$

and the control law is:

$$\dot{\mathbf{r}} = -\lambda\widehat{\mathbf{e}} \tag{8}$$

Considering equation (4), this control law is equivalent to a weighted mean of the control laws of each sub-system $\dot{\mathbf{r}}_i = -\lambda\widehat{\mathbf{e}}_i$:

$$\dot{\mathbf{r}} = -\lambda\sum_{i=1}^{N}\kappa_i\widehat{\mathbf{e}}_i = \sum_{i=1}^{N}\kappa_i\dot{\mathbf{r}}_i \tag{9}$$

5 Stability analysis

In order 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 case of 2D visual servoing, it is simply $\widehat{\mathbf{e}} = \mathbf{e}$ if the measurement noise is neglected. The closed-loop system with the control law (8) is thus:

$$\dot{\mathbf{e}} = -\lambda(\mathbf{CLW})(\mathbf{C}\widehat{\mathbf{L}}\widehat{\mathbf{W}})^{-1}\mathbf{e} = -\lambda(\widehat{\mathbf{L}}\widehat{\mathbf{W}})^\sharp(\mathbf{LW})\mathbf{e} \tag{10}$$

A sufficient condition for the stability of the system is:

$$(\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^\sharp\mathbf{LW} > 0 \tag{11}$$

It is clear now why it is interesting to choose $\mathbf{C}$ as a block pseudo-inverse. Indeed, using equation (3), the condition (11) can be also written as:

$$(\widehat{\mathbf{L}}^*\widehat{\mathbf{W}})^\sharp\mathbf{LW} = \sum_{i=1}^{N}\kappa_i\widehat{\mathbf{W}}_i^{-1}\widehat{\mathbf{L}}_i^{*+}\mathbf{L}_i\mathbf{W}_i > 0 \tag{12}$$

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

$$\widehat{\mathbf{W}}_i^{-1}\widehat{\mathbf{L}}_i^{*+}\mathbf{L}_i\mathbf{W}_i > 0 \qquad \forall i = \{1, 2, 3, ..., N\} \tag{13}$$

The whole system will be thus stable if each sub-system 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 sub-system separately. Using the control law (9) the norm of $\mathbf{e}$ decrease at each iteration. Local minima could occur when $\mathbf{e} = 0$ even if $\mathbf{e}_i \neq 0$. However, it is easy to detect and eliminate these local minima since at the convergence $\mathbf{e} = 0 \ \forall\kappa_i$. Any change in the weighting factors κ_i will bring the system away from the local minimum.

Unfortunately, the stability domain of the 2D visual servoing is not analytically known to this date and the analyse have to be stopped. However, a 2D mono-camera visual servoing system is known to be robust to calibration errors [3]. The same robustness can thus be expected from a multi-cameras system using a 2D visual servoing technique.

In the case of the $2\frac{1}{2}$D visual servoing the stability analysis is a little more complex since, for each sub-system, we have $\widehat{\mathbf{e}}_i = \mathbf{E}_i \mathbf{e}_i$ [8]. Then, the estimated task function is:

$$\widehat{\mathbf{e}} = \sum_{i=1}^{N} \kappa_i \mathbf{E}_i \mathbf{e}_i \qquad (14)$$

Then, $\widehat{\mathbf{e}}$ cannot be written easily as a function of $\mathbf{e}$ under the form $\widehat{\mathbf{e}} = \mathbf{E}\,\mathbf{e}$ as in the mono-camera visual servoing. However, we can proceed in the following way. The task function can be written as:

$$\mathbf{e} = \begin{bmatrix} \kappa_1 \mathbf{I}_6 & \kappa_2 \mathbf{I}_6 & \dots & \kappa_N \mathbf{I}_6 \end{bmatrix} \mathbf{e}' = \mathbf{K} \mathbf{e}' \qquad (15)$$

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

$$\mathbf{e}' = \mathbf{K}^{+} \mathbf{e} + (\mathbf{I}_{6N} - \mathbf{K}^{+}\mathbf{K})\phi \qquad (16)$$

where $(\mathbf{I}_{6N} - \mathbf{K}^{+}\mathbf{K})\phi$ is a vector (whose form is not important) belonging to the null space of $\mathbf{K}$. Thanks to the simple form of $\mathbf{K}$, is $\mathbf{K}^{+} = (\sum_{i=1}^{N} \kappa_i^2)^{-1}\mathbf{K}^T$. Plugging equations (16) and (15) in equation (14), the estimated task function is:

$$\widehat{\mathbf{e}} = \frac{1}{\sum_{i=1}^{N} \kappa_i^2} \sum_{i=1}^{N} \kappa_i^2 \mathbf{E}_i\, \mathbf{e} + \mathbf{b}_e = \mathbf{E}\,\mathbf{e} + \mathbf{b}_e \qquad (17)$$

where $\mathbf{b}_e$, which can be considered as an additional "bias" on the task function if the calibration errors are not too big, is:

$$\mathbf{b}_e = \sum_{i=1}^{N} \kappa_i \mathbf{E}_i \phi_i - \frac{1}{\sum_{i=1}^{N} \kappa_i^2} \sum_{i=1}^{N} \kappa_i^2 \mathbf{E}_i (\sum_{i=1}^{N} \kappa_i \phi_i) \quad (18)$$

An approximation can be made and the bias can be neglected. This hypothesis is justified since, thanks to the very nice form of matrix $\mathbf{E}_i$ (block diagonal), and with an appropriate choice of $\mathbf{e}_i$ [7], then $\mathbf{E}_i \approx \mathbf{I}$ and the "bias" $\mathbf{b}_e \approx 0$. Let us remark that at the convergence $\mathbf{E}_i = \mathbf{I}$ then $\mathbf{b}_e = 0$ and the system is always locally stable [7]. Furthermore, thanks to the nice form of matrices $\mathbf{L}_i$ and $\mathbf{W}_i$ (block triangular), the stability analysis of the multi-cameras $2D\frac{1}{2}$ visual servoing is again reduced to the stability analysis of each sub-system. Since each sub-system is stable under the conditions given in [8], the whole system is also stable, at least under the same conditions.

Depending on the stability domain which is possible to obtain from the control law, two types of algorithms can be considered. When the convergence domain of each sub-system is not large (as for the 2D visual servoing), the following algorithm is proposed:

1) Test how many cameras have converged (a camera has converged if the error on visual features is lower than ϵ_i: $\|\mathbf{s}_i - \mathbf{s}_i^*\| < \epsilon_i$). If all the cameras have converged the goal is reached, else go to 2).

2) Select all the converged cameras more the nearest to convergence (the nearest camera to convergence is the one minimising $\|\mathbf{s}_j - \mathbf{s}_j^*\|$ under the constraint that $\|\mathbf{s}_j - \mathbf{s}_j^*\| > \epsilon_j$). Go to 3).

3) Do visual servoing using the selected cameras and after their convergence, go to 1).

This iterative algorithm add a new camera at each step. If the stability domain is very large (as for the $2\frac{1}{2}$D visual servoing), a more satisfactory algorithm can be considered:

1) Test how many cameras have converged. If all have converged the goal is reached, else go to 2).

2) Select all the converged cameras, more those which have the target in their field of view (and which are in the stability domain). Go to 3).

3) Do visual servoing using the selected cameras and after their convergence go to 1).

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

- a converged camera has also converged at the end of each step. This does not means 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, i.e. with the target in its field of view and in the stability domain.

6 Experimental Results

A system simulating the positioning of the cover has been tested. In order to demonstrate the practicability of the task, only two cameras were used. The cameras are placed at the extremity of a 50 cm aluminium bar, mounted on a manipulator robot as in Figure 2. The Mitsubishi PA10 robot at the EDF-DER was used as well as two planar targets with six points each. A simple hardware architecture was used in order to reduce the cost of the system: the two cameras are connected to two different entry of the same video-board and the acquisition of the two images is sequential. The rate of such system is two times slower than 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 could not 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.
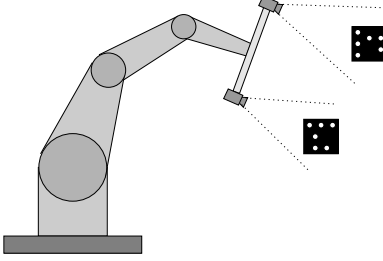

Figure 2: Positioning a cumbersome objet

6.1 Precision comparison between mono and multi-cameras visual servoing

position	1	2	3	4
translation	31 %	56 %	46 %	48 %
rotation	36 %	58 %	27 %	44 %

Table 1: Positioning precision improvements

In these first experiments, 2D visual servoing is used. The system is robust with respect to the calibration errors since a teaching by showing technique is used. The 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 where mainly rotations around the $\overrightarrow{x}$ and the $\overrightarrow{y}$ axes respectively. Finally, the fourth displacement was a big rotation around the $\overrightarrow{x}$ and $\overrightarrow{y}$ axes and a big translation. First, the visual servoing is carried out using only one of the two cameras. Then, starting to the same initial position, the servoing is performed using the two cameras with the control law (8). In all the experiments, the positioning precision is considerably improved using the multi-cameras visual servoing as it is shown in the Table 1 (typically from 1.98 mm to 0.86 mm). These results demonstrate the interest of using several cameras to position a cumbersome object.

6.2 Comparison between 2D and $2\frac{1}{2}$D multi-cameras visual servoings

Even if the multi-cameras visual servoing can be used with any scheme (2D, $2\frac{1}{2}$D and 3D) we compare here only the 2D and $2\frac{1}{2}$D methods whose results are illustrated in Figure 3 and 4 respectively. The figures are divided in two columns, one for each camera.

(a) left coord. errors   (b) right coord. errors

(c) left points trajectory   (d) right points trajectory

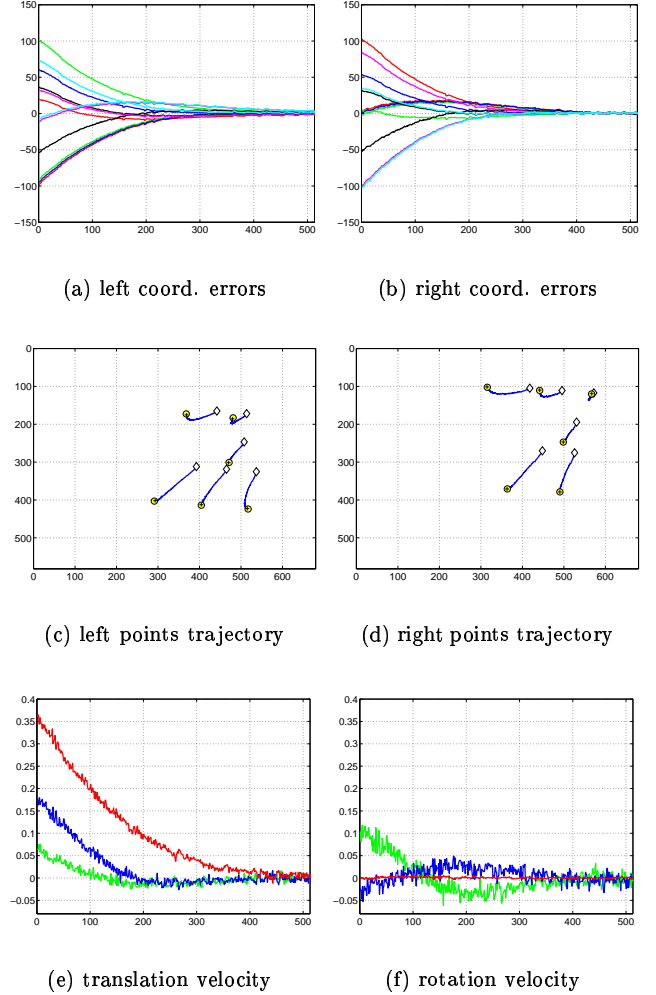(e) translation velocity   (f) rotation velocity

Figure 3: 2D visual servoing results

In each column we give respectively the error on the coordinates of the points, their trajectory in the image (diamond and circle marks correspond respectively to the initial and final positions). In the last row, we give the control law (i.e. the velocity of the robot end-effector). The error on the coordinates goes to zero and the control law is stable even with the imprecise calibration that was used. Indeed, the camera intrinsic parameters and the transformation matrices between each camera and end-effector frame were only roughly

known. The speed of convergence is relatively slow at the end of the servoing since a simple proportional control law was used. It can be easily improved by simply increasing gain λ as soon as the error becomes small.

(a) left coord. errors

(b) right coord. errors

(c) left points trajectory

(d) right points trajectory

(e) translation velocity
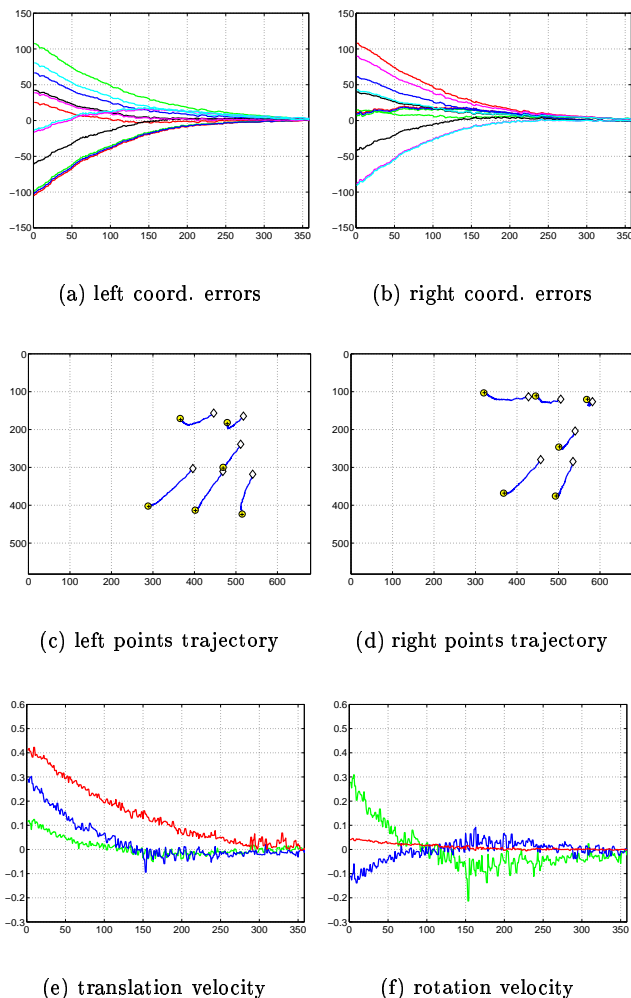
(f) rotation velocity

Figure 4: $2\frac{1}{2}D$ visual servoing results

The trajectories of the points in the images are quite the same (even if the control law is different) since the displacement was not very large. Experiments with bigger displacements were also realised but only one target was initially in the image. In this case, we initially perform the visual servoing using only one camera and once the two target are in the images, we perform the multi-cameras visual servoing. The positioning precision measured for the 2D and $2D\frac{1}{2}$ methods are the same, since they are independent on the used scheme but uniquely on the maximal error reached in the image (the servoing is stopped when the maximal error on the coordinates of the image points is 1 pixel).

7   Conclusion

In this paper, a multi-cameras visual servoing was proposed within the task function framework. Our visual servoing scheme uses several cameras observing different parts of a target. The particular choice of the task function simplifies the design of the control law and the stability analysis. Indeed, the stability analysis of the whole system is ensured by the stability of each sub-system. The experiments, carried out with 2D and $2\frac{1}{2}D$ visual servoing, validate the generality of our method, and the possibility of its extension, if needed, to the 3D scheme.

Acknowledgements

References

[1] P. K. Allen, A. Timcenlo, B. Yoshimi, and P. Michelman. Hand-eye coordination for robotic tracking and grasping. In K. Hashimoto, editor, *Visual servoing*, pages 33–69. World Scientific Press, 1993.

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

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

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

[5] G. D. Hager and Z. Dodds. A projective framework for constructing accurate hand-eye systems. In *Work. on New Trends In Image-Based Robot Servoing (IROS'97)*, pp. 71–82, Grenoble, France, September 1997.

[6] K. Hosoda and M. Asada. Versatile visual servoing without knowledge of true jacobian. In *IEEE IROS*, pages 186–193, September 1994.

[7] E. Malis. *Contributions à la modélisation et à la commande en asservissement visuel*. PhD thesis, Université de Rennes I, IRISA, November 1998.

[8] E. Malis, F. Chaumette, and S. Boudet. 2 1/2 d visual servoing. *IEEE Trans. on Robotics and Automation*, 15(2):234–246, April 1999.

[9] N. Maru, S. Y. H. Kase, A. Nishikawa, and F. Miyazaki. Manipulator control by visual servoing with the stereo vision. In *IEEE IROS*, volume 3, pages 1866–1870, Yokohama, Japan, July 1993.

[10] A. Ruf and H. R. Rigid and articulated motion seen with an uncalibrated stereo rig. In *IEEE ICCV*, Corfou, Greece, September 1999.