# One Click Focus with Eye-in-hand/Eye-to-hand Cooperation

Claire Dune, Eric Marchand, Cédric Leroux

# One Click Focus with Eye-in-hand/Eye-to-hand Cooperation

Claire Dune[1,2], Eric Marchand[1], Christophe Leroux[2]

***Abstract*— A critical assumption of many multi-view control systems is the initial visibility of the regions of interest from all the views. An initialization step is proposed for a hybrid eye-in-hand/eye-to-hand grasping system to overcome this requirement. In this paper, the object of interest is assumed to be within the eye-to-hand field of view, whereas it may not be within the eye-in-hand one. The object model is unknown and no database is used. The object lies in a complex scene with a cluttered background. A method to automatically focus the object of interest is presented, tested and validated on a multi view robotic system.**

## I. INTRODUCTION

Service robotics is a fast growing field. Among its applications are robotic assistants for elderly or disabled people. They can provie these persons with of recovering some manipulation capabilities in everyday-life. Since their handicap often prevents the disabled from moving their arms and legs, the action necessary to command such robotic assistants has to be strictly limited. In this paper, we propose the first step towards a semi-autonomous application for a grasping task which only requires one click from the user.

The robotic system consists of two cameras. One camera is fixed in the workspace (typically this is a wideangle camera able to see the whole working area and attached to the top of a wheelchair). The other camera is mounted on a robot end-effector and is of limited sight. It can move close to the scene and is able to capture scene details. The goal is to achieve eye-in-hand/eye-to-hand cooperation with these two cameras. Few papers [10], [7], [5] (and to some extent [9]) deal with eye-in-hand/eye-to-hand cooperation.

[10], [7], [5] and most of the multi camera systems assume that the interest area is common to every cameras field of view. On the contrary, in the proposed approach, the object of interest is within the filed of view of only the eye-to-hand camera. This assumption is not necessarily for the mobile camera since the initial arm pose is assumed to be purely random. We propose an approach to semi-automatically set the mobile camera's position so as to center the interest area in its field of view. The object is assumed to be unknown, i.e. its shape and texture are not known, no database is used, and the scene is complex with a textured and cluttered background. Furthermore, the object can move slightly within the scene. The hybrid system is supposed to be calibrated.

[1] INRIA, IRISA, Lagadic Project, F-35000 Rennes, France
[2] CEA List, F-92265 Fontenay Aux Roses, France

This work has been done at IRISA-INRIA Rennes in collaboration with CEA List. It was also supported by CEA and by Brittany County Council under contribution to student grant.

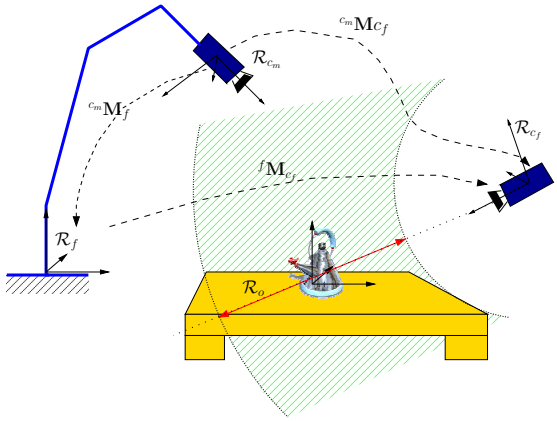


Fig. 1. Eye-in-hand/eye-to-hand system reference frames

The application starts as soon as the user has clicked on the object to grasp in the eye-to-hand image. Due to the epipolar geometry constraint [8], the corresponding point is known to be on a line in the eye-in-hand image. A visual servoing scheme based on epipolar geometry allows the eye-in-hand camera to center the line in its image. The camera can then be controlled to cover the line while the centering task ensures that the object will be in the eye-in-hand field of view at some moments. Epipolar-geometry-based robot control has been studied in the past, mostly for visual homing applications [12], [13], [1]. In [12], [13] the visual servoing is based on the matching of a desired epipole position and the current computed epipole. In [1] the epipolar lines in both the current and the desired views are computed and aligned. These methods assume that some common features are observed in both current and desired views to estimate the epipolar geometry. In [9] an initialization step is used to ensure that the object detected by the fixed camera falls within the mobile camera's field of view. Our approach can be seen as an extension of [1] and [9] as the servoing task consists of surfing the epipolar line looking for the object of interest. Similar to [9], no image is acquired at the desired camera position. The system is calibrated and the epipolar geometry is not estimated but explicitly computed. In [9] the baseline of the system is fixed and small, whereas the considered mobile camera is mounted on a six degrees of freedom (dof) arm and is far from the eye-to-hand camera. Besides, in [9], a geometric and kinematic coupling between both cameras is built, while we propose to use a visual servoing scheme [6].

Section *II* is dedicated to the epipolar geometry based visual servoing. A complete solution to ensure that the

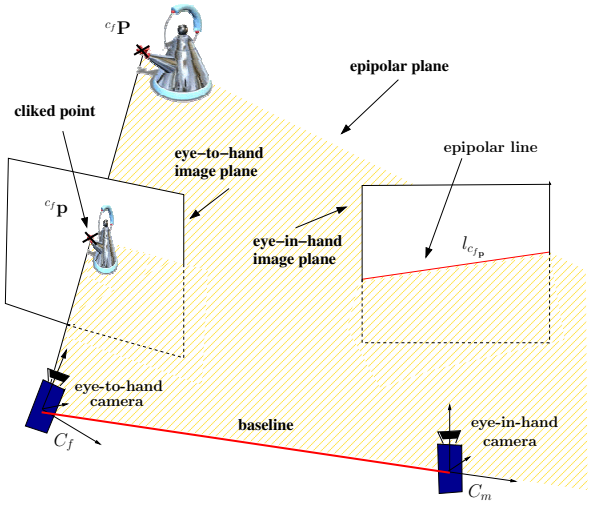


Fig. 2. eye-in-hand/eye-to-hand system reference frames

object of interest is centered in the mobile camera's view is presented. Section *III* focuses on the search of the object using Lowe's SIFT point matching and Bayesian decision framework. Section *IV* presents some experimental results that validate the proposed method.

## II. Epipolar geometry based visual servoing

In this section, the eye-in-hand/eye-to-hand cooperation framework to control the centering of the line of view and its exploration are presented.

### A. Eye-in-hand/eye-to-hand system

Let $\mathscr{R}_f$, be the system reference frame. Consider $c_f$, a fixed camera, and $c_m$, a mobile camera mounted on the robot end-effector (see Figure 1). Let $\mathscr{R}_{c_f}$ and $\mathscr{R}_{c_m}$ be the fixed camera and the mobile camera frame respectively. $\mathscr{R}_o$ is the object frame. ${}^{f}\mathbf{M}_{c_m}$ and ${}^{f}\mathbf{M}_{c_f}$ are the transformations between the cameras and the base frame. Finally, let ${}^{c_f}\mathbf{M}_{c_m}$ be the transformation between the two cameras. ${}^{f}\mathbf{M}_{c_f}$ is constant (and can be evaluated through a classical pose estimation procedure), the robot odometry gives a precise estimation of ${}^{c_m}\mathbf{M}_{f}$ at every stage. ${}^{c_m}\mathbf{M}_{c_f}$ can thus be computed by ${}^{c_m}\mathbf{M}_{c_f} = {}^{c_m}\mathbf{M}_{f}\,{}^{f}\mathbf{M}_{c_f}$.

### B. Multi-view Geometry

Let $\mathbf{C}_m$ and $\mathbf{C}_f$ be the two optical centers of $c_m$ and $c_f$ respectively. The line $\mathbf{C}_f\mathbf{C}_m$ is called the baseline (see Figure 2). The epipolar geometry sets the relation between a 3D point ${}^{c_f}\mathbf{P}$ and its projections in the two camera image planes ${}^{c_f}\mathbf{p}$ and ${}^{c_m}\mathbf{p}$. The epipolar geometry constraint can be written as [8]:

$$ {}^{c_f}\mathbf{p}^{T}\,{}^{c_f}\mathbf{E}_{c_m}\,{}^{c_m}\mathbf{p} = 0 \tag{1}$$

where ${}^{c_f}\mathbf{E}_{c_m}$ is called the essential matrix and is defined by:

$$ {}^{c_f}\mathbf{E}_{c_m} = [{}^{c_m}\mathbf{t}_{c_f}]_{\times}\,{}^{c_m}\mathbf{R}_{c_f} \tag{2}$$

with ${}^{c_m}\mathbf{t}_{c_f}$ and ${}^{c_m}\mathbf{R}_{c_f}$ the translation and rotation matrix between the two frames and $[.]_{\times}$ the cross product.

The essential matrix (2) defines a relation between the epipolar constraint and the extrinsic parameters of the system. Since ${}^{c_m}\mathbf{M}_{c_f}$ is given, the essential matrix can be computed at every stage. Furthermore, in (1), ${}^{c_f}\mathbf{E}_{c_m}\,{}^{c_m}\mathbf{p}$ is the epipolar line that goes through ${}^{c_f}\mathbf{p}$ and the epipole ${}^{c_m}\mathbf{e}$ (the projection of $\mathbf{C}_f$ on $c_m$ image plane). The epipolar line is the projection of the line of view corresponding to ${}^{c_f}\mathbf{p}$. The essential matrix is therefore the mapping between points and epipolar lines we were looking for. Since our system has a wide baseline, epipolar geometry is stable and it can be used robustly to perform 3D reconstruction. Knowing the point ${}^{c_f}\mathbf{p}$ and the calibration of the system, the epipolar line equation in the eye-in-hand image plane can be deduced regardless of potential motion of ${}^{c_f}\mathbf{p}$. In the following, the epipolar line is represented by the two parameters $(\rho,\theta)$. The line equation under this representation is the following :

$$ x\cos\theta + y\sin\theta = \rho \tag{3}$$

$\rho$ and $\theta$ are computed from equation (1).

### C. Visual servoing based on the epipolar line

We now present the visual servoing control scheme. Let us first recall the task specifications: the epipolar line associated with ${}^{c_f}\mathbf{p}$ has to be brought in the mobile view. Ensuring this line remains horizontal and centered, the mobile camera moves along it to search for the object.

*1) Focus on the epipolar line:* Visual servoing is a robotic control based on visual features extracted from one or several cameras. Let $\mathbf{s}$ be the current visual feature and $\mathbf{s}^*$ be the desired visual feature. The main task is to regulate the error vector $\mathbf{e_1}=\mathbf{s}-\mathbf{s}^*$ to zero. The associated interaction matrix $\mathbf{L}_1$ of the task $\mathbf{e_1}$ links the time variation of the selected visual features to the relative camera object kinematics screw. It is defined by [6]:

$$ \dot{\mathbf{s}} = \mathbf{L}_s\mathbf{v} \tag{4}$$

Then, considering a eye-in-hand camera the control law that regulates $\mathbf{e_1}$ is [6]:

$$ \mathbf{v} = -\lambda_1\widehat{\mathbf{L_1}}^{+}\mathbf{e_1} + \mathbf{P}\mathbf{z} \tag{5}$$

Where $\widehat{\mathbf{L_1}}^{+}$ denotes the pseudo inverse of an approximation or a model of $\mathbf{L_1}$, $\mathbf{z}$ is an arbitrary secondary control vector and $\mathbf{P} = \mathbf{I} - \mathbf{L_1}^{+}\mathbf{L_1}$ is a projection operator that guarantees that the control vector $\mathbf{z}$ has no effect on the main task $\mathbf{e}_1$. Let us introduce a secondary task $\mathbf{e_2}$ and its associated matrix $\mathbf{L_2}$. $\lambda_1$ is a positive gain that tunes the exponential decrease of the task. Then $\mathbf{z}$ is set to be:

$$ \mathbf{z} = -\lambda_2\widehat{\mathbf{L_2}}^{+}\mathbf{e_2} \tag{6}$$

By including (6) in (5), the control law computed from the two tasks is:

$$ \mathbf{v} = -\lambda_1\widehat{\mathbf{L_1}}^{+}\mathbf{e_1} - \lambda_2\mathbf{P}\widehat{\mathbf{L_2}}^{+}\mathbf{e_2} \tag{7}$$

In our case the primary task is a focusing task w.r.t the epipolar line while the secondary task allows movement along the line.

*2) Primary task: centering the epipolar line:* Since the feature $\mathbf{s} = (\rho, \theta)$ is a line, its associated interaction matrix is given by [6]:

$$\mathbf{L_1} = \begin{pmatrix} \lambda_\rho c\theta & \lambda_\rho s\theta & -\lambda_\rho \rho & (1+\rho^2)s\theta & -(1+\rho^2)c\theta & 0 \\ \lambda_\theta c\theta & \lambda_\theta s\theta & -\lambda_\theta \rho & -\rho c\theta & -\rho s\theta & -1 \end{pmatrix} \quad (8)$$

with $s\theta = \sin(\theta)$ and $c\theta = \cos(\theta)$. $\lambda_\rho$ and $\lambda_\theta$ are given by:

$$\begin{cases} \lambda_\rho &= (a\rho\cos\theta + b\rho\sin\theta + c)/d \\ \lambda_\theta &= (a\sin\theta - b\cos\theta)/d \end{cases} \quad (9)$$

where $aX+bY+cZ+d=0$ is an equation, expressed in $\mathscr{R}_{c_m}$, of the plane that contains $\mathbf{C}_f$, ${}^{c_f}\mathbf{p}$ and which is perpendicular to the epipolar plane.

When the visual servoing task ensures the epipolar line is horizontal and centered, a secondary task can be considered to look along this line for the object of interest.

If an open loop was considered, it would have been impossible to handle potential motion of the object. Visual servoing can overcome this problem, since the features are updated at each step of the control and it is robust to object motion.

*3) Secondary task: covering the epipolar line:* The search of the object is limited to the segment which is the intersection of the epipolar line and the robot workspace. The extremities are two 3D points ${}^{c_f}\mathbf{P}_i = (X_i, Y_i, Z_i)$, $i \in \{1,2\}$, in $\mathscr{R}_{c_f}$. They are projected on the image plane of the mobile camera as $\mathbf{p_i} = (x_i, y_i)$, $i \in \{1,2\}$.

Since the centering of the epipolar line constraints two DoF, four DoF are available for the secondary task. Using the redundancy formalism as presented in Section II-C.1, centering of the ${}^{c_f}\mathbf{P}_i$ can be achieved without disturbing the regulation of the primary task.

The centering of the point $\mathbf{P}_i$ in the mobile field of view is achieved by using a secondary task $\mathbf{e_2}$. Its regulation aims at bringing the 2D point $\mathbf{p}_i$ on the point $\mathbf{p}^* = (0,0)$ : $\mathbf{e_2} = \mathbf{p}_1 - \mathbf{p}^*$. The associated interaction matrix $\mathbf{L_2}$ is the matrix related to a 2D point [6].

Considering an open loop control, if the line centering is not perfect, rotating the mobile camera along its y axis increases the error and the line is no longer guaranteed to be inside the mobile camera's field of view. The use of redundancy overcomes this problem. It ensures that the secondary task does not disturb the achievement of the primary task.

Visual servoing based on epipolar geometry provides a way of controlling an eye-in-hand camera to move along the line of view that stems from a selected 2D point in the eye-to-hand image plane. At this point the mobile camera is ensured to run along the 3D segment of interest. The next section is dedicated to searching the object along the epipolar line.

## III. Localization of the object on the epipolar line

The object to grasp projects in the neighborhood of the point ${}^{c_f}\mathbf{p}$ in the eye-to-hand image. So, a view of the object is available and can be used to detect the object in the eye-in-hand image while the eye-in-hand camera is covering the epipolar line associated to ${}^{c_f}\mathbf{p}$.

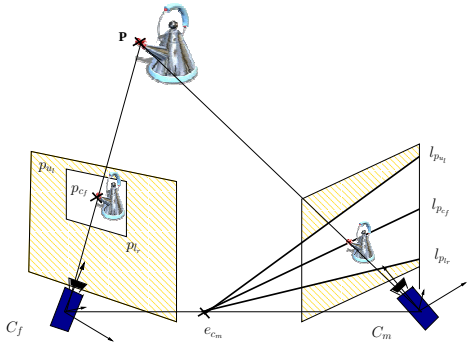


Fig. 3. Selection of the interest areas

A classical object recognition schemes, such as Lowe's SIFT [11], can be used to match the appearance of the object in eye-to-hand and eye-in-hand images. The main assumption of the presented approach is that more features are found on the object area than on the rest of the segment giving information on the object localization. The Lowe's features [11] are famous for their invariance properties, to image scale and rotation, and provide robust matching across affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. Yet, since it is a multi-scale method, it is time-consuming.

To reduce computational time, the feature extraction is limited to a region of interest in both views. In the $c_f$ view, the extraction is restricted to the neighborhood of ${}^{c_f}\mathbf{p}$. In [2] it is assumed that the regions of high edge densities are candidates for the object. These regions are highlighted as blobs of interest. The blob surrounding a user selected point is kept. Active edge detector, level sets, intrinsic scale or growing region methods can also be used to detect the object area. The computed object area contains a pattern made of at least a part of the object and some of the surrounding background. That pattern is the reference for the recognition scheme. In the eye-in-hand image, the search of the object is limited to the area between the two extremal epipolar lines corresponding to the selected area in the eye-to-hand image (see Figure 3).

The matching process thus consists of three main steps: 1) select the interest areas in the two views 2) extract the Lowe's SIFT [11] 3) use Lowe's SIFT [11] matching to detect the object in the eye-in-hand image set.

### A. Projection of the features matched on the 3D segment

The object of interest is now on the epipolar line. In this part, we propose a solution to estimate its depth $D$. During the mobile camera's motion, SIFT features are extracted and matched with the pattern selected in the fixed view.

Thanks to the robustness and the accuracy of visual servoing, we can assume that the epipolar line is horizontal and centered and therefore only consider the x-coordinates of the matched features. Then, the depth of features projection on the 3D line is calculated using (4).

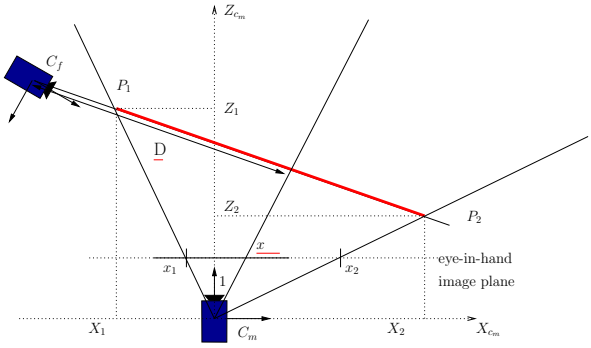


Fig. 4. Top view of the system: given the x coordinate in the mobile view, compute the depth of the point on the 3D segment.

Consider the ends of the segment ${}^{c_m}\mathbf{P_i}$. As soon as the primary task is regulated, the epipolar plane is of coordinates (0,1,0,0). Besides, $\mathbf{C}_m$, and both the 3D points ${}^{c_m}\mathbf{P_i}$ are on the epipolar plane. Therefore the $Y_i$ coordinates are null. Let $(x,y)$ be the coordinate of a feature point ${}^{c_f}k$ and let $(X,0,Z)$ be the coordinates of its projection on the segment.

$$\begin{aligned} X &= X_1 + (X_2 - X_1)D/L \\ Z &= Z_1 + (Z_2 - Z_1)D/L \end{aligned} \tag{10}$$

where $L = \sqrt{(X_1 - X_2)^2 + (Z_1 - Z_2)^2}$ and $D$ is the depth associate to $x$ on the 3D segment.

From (10), $D$ is deduced:

$$D = L\frac{xZ_1 - X_1}{(X_2 - X_1) - x(Z_2 - Z_1)} \tag{11}$$

This equation sets the relation between a feature ${}^{c_f}k$ in the mobile image plane and its depth if it was part of an object lying on the 3D segment. To model the measurement error, the positions on the 3D segment are not represented as Dirac functions but as Gaussian functions $(\mu,\sigma)$, where $\mu$ is set to D, the projection of the feature on the segment and $\sigma$ represents the uncertainty in the depth estimation.

Let $k_{ii\in 1..N}$, be a set of $N$ matched points extracted from a view. Let $D_{ii\in 1..N}$, be their relative depth, then:

$$\widehat{D} = max\Big\{\sum_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{1}{2}(\frac{x-\mu_i}{\sigma})^2}\Big\} \tag{12}$$

Each set of matched features extracted from views of the 3D segment gives an estimation of the object depth $\widehat{D}$. If no feature is matched, the object is likely to be on the part of the epipolar line that is not seen.

### *B. Bayesian decision framework*

Finding the depth of an object lying on a line has been investigated over the past few years mainly for objects localization in the mobile robot context and more precisely in the initialization of the feature depth in Simultaneous Localization and Mapping issue [3]. A feature depth is estimated using Bayesian inference over a set of successive views where the tracked features lie. On the contrary, our approach does not imply that the object is seen in all the views taken along the segment. At each step of the decision process the unseen part of the segment has to be handled. Decision process is then used to determine the depth of the object on the epipolar line (Figure 5).

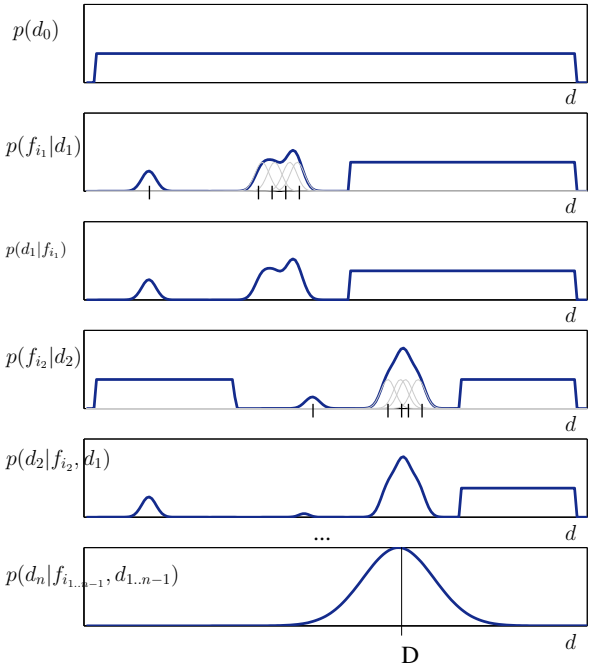


Fig. 5. Bayesian decision process: 1) The top frame represents the prior knowledge at the initialization step. The probability density function (pdf) is uniform between the minimum and maximum depth and null elsewhere. 2) The second frame is the pdf corresponding to the first measurements. Some features have been matched and their projection on the 3D line are represented by Gaussian function. 3) using Bayesian equation (13) the posterior is calculated. It is used as a prior for the next step. 4) A new set of measurements is taken given a new pdf. 5) Posterior is computed using the pdf of the third and fourth frame, and so on, until the pdf converges to a Gaussian distribution that has a maximum at the object estimated depth.

In the beginning, there is no knowledge on the object depth. The a priori probability density function (we refer to it as 'the prior') is therefore uniform on the segment and zero elsewhere. At each step of the chaining, a likelihood is computed based on the feature detection. This likelihood, together with the prior is used to compute the posterior, through a Bayesian rule [4]. The chaining is done by using the posterior of a step as the prior for the next step.

Let $k_i$ be a set of $i$ features, $D$ the depth of the object, $L$ the length of the segment and $p(x)$ a probability density function, Bayes theorem gives:

$$p(D_{t+1}|k_{i_{t+1}}, D_{0..t}) = \frac{p(D_t|k_{i_t}, p(D_{0..t-1}))p(k_{i_{t+1}}|D_{t+1})}{\int_L p(D_t|k_{i_t}, p(D_{0..t-1}))p(k_{i_{t+1}}|D_{t+1})dD} \tag{13}$$

As soon as the segment has been entirely seen, the depth estimation is computed as the maximum of the a posteriori probability. To refine the estimation, the segment can be covered several times. Two stop criteria can then be used: a

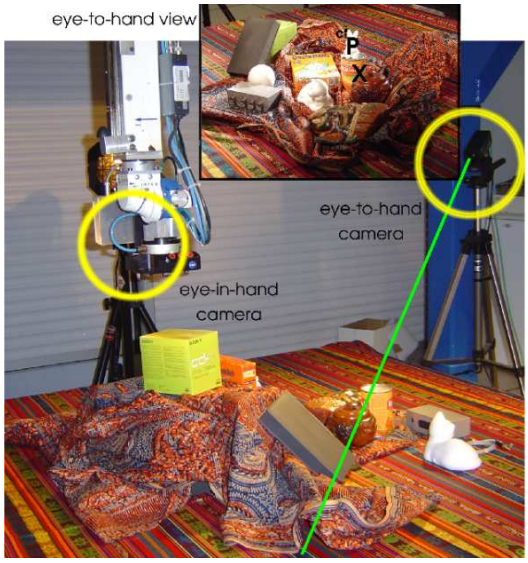


Fig. 6. Experimental setup: the scene is complex and the background is textured. The two camera locations are highlighted. Top right, the eye-To-Hand view is display with the clicked point ${}^{c_f}\mathbf{p}$. The green line represents the line of view associated with ${}^{c_f}\mathbf{p}$

threshold on the maximum posterior value or a measure of the information, like Shannon's entropy. A compromise has to be made between the accuracy of the estimation and time spent to compute it.

As soon as the stop criterion is reached, an estimation of the object depth $\widehat{D}$ is returned.

## IV. Experimental results

This section presents a typical execution of the application presented above. The experimental setup is presented in Figure 1 and 6. The mobile camera is mounted on the end effector of a 6-DoF arm robot. The eye-to-hand camera is fixed and its field of view covers the whole robot workspace. As figure 6 shows, the scene is quite complex and the background is highly textured. The algorithm is launched by clicking the image of the object of interest in the fixed view.

First, the epipolar line is centered. When the main task error falls below a certain threshold the secondary task is activated and the 3D segment is scanned. While the eye-in-hand camera is covering the segment, the object is searched in the mobile view. The mobile camera keeps moving until the object is found.

We first present the results of the visual based control scheme and then the results of the detection process.

### A. Epipolar based visual servoing

The control scheme results are summed up in Figures 7, 8 and 9. Figure 7 shows the evolution of the two tasks $\mathbf{e_1}$ and $\mathbf{e_2}$ during the regulation. The execution starts with only the main task. Then, at iteration 30, the main task error passes below a fixed threshold and the secondary task is launched. The point to center is the projection of $\mathbf{P}_1$, the first extremity of the 3D segment. The secondary task error decreases while the main task error remains zero. At iteration 60, segment extremity $\mathbf{P}_1$ is reached; the point to center is then $\mathbf{P}_2$. The secondary task error increases suddenly when the referenced point is changed, and is then regulated on an exponential decrease. Figure 8 gives the robot-end-effector velocities. The execution of the secondary task at iterations 30 and 60 implies, as expected, a pure rotational motion around the $y$ axis of the $\mathscr{R}_{c_m}$ frame.

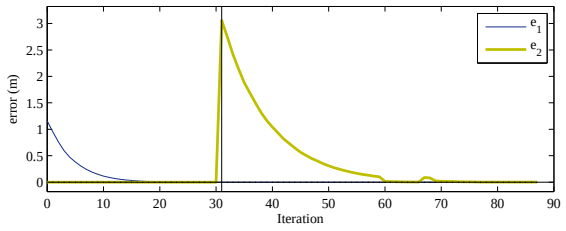


Fig. 7. Task error during a visual servoing execution on a motionless target

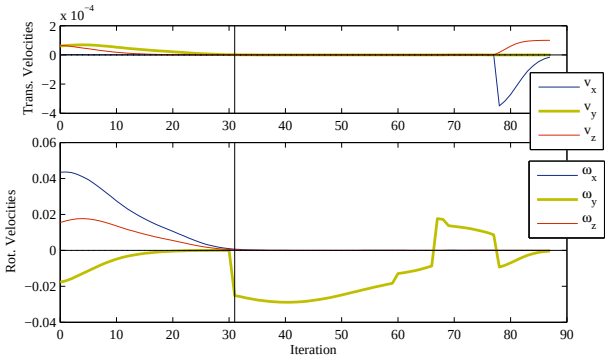


Fig. 8. Velocities of the mobile camera during the servoing on a motionless target

We performed another experiment (Figure 9) with a moving object to validate the robustness of the proposed control scheme. A simple tracking algorithm based on local appearance gives the coordinates of ${}^{c_f}\mathbf{p}$ at each step of the servo loop, so that we can compute each time the epipolar line and the points $\mathbf{P}_i$. The control law thus takes into account the movement of the object.

At the beginning of the experiment, when the main task is launched, the targeted object is motionless. The object starts moving at iteration 90 while the main task is not completed. It stops moving at iteration 190, and then moves again from iteration 230 to 330. The secondary task is launched at iteration 160 when the main task is completed.

The main task is hardly disturbed by the object motion. As can be seen in Figure 9(c), the main task error is a perfect exponential before iteration 160, and is perfectly regulated afterwards. The secondary task is disturbed more by the object motion. The error decreases, but is not regulated to zero. A tracking error can be clearly noticed from iteration 200 to 330 (see Figure 9). This error is quickly corrected as soon as the object stops.

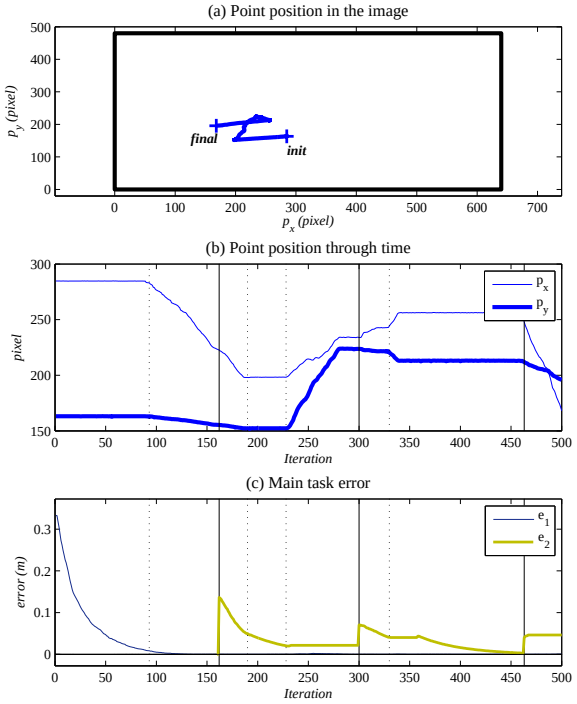


Fig. 9. Visual servoing on a mobile target: the top frame illustrates the movement of ${}^{c_f}\mathbf{p}$ in the eye-in-hand image plan. The middle frame represents the evolution of the coordinates of ${}^{c_f}\mathbf{p}$ over time. The bottom frame is the evolution of the task error

### B. Searching for the object

As soon as the segment is in the eye-in-hand view, the recognition algorithm starts and the object is searched. SIFT features are extracted from eye-in-hand views and matched with the features extracted from the region of interest of the fixed camera view. The likelihood of the object depth on the line of view is then computed using the projection of the matched features on the 3D segment. The posterior is computed using (13). The process iterates until the maximum posterior reaches 0.5. Figure 10 presents the evolution of the a posteriori probability density function of the object depth. A clear maximum quickly appears and the object is easily found within 10 iterations of the recognition process in which the segment is covered twice. The object is finally found in the camera view. Its depth on the line of view is approximately $1,37m$.

## V. Conclusion

We have investigated an eye-to-hand/eye-in-hand cooperation scheme, that works as an initialization tool for an active multi-camera system. It provides the first step towards a one-click-configured semi-autonomous grasping task with no a priori knowledge about the object.

Thanks to the proposed method, the mobile camera automatically moves until the object of interest falls in its field of view. The algorithm is robust to some slight motions of the targeted object since the searching area is updated at each step of the servo process. The estimation of the object localization is achieved using a Bayesian framework. The accuracy of the object position can be refined by covering the segment several times.

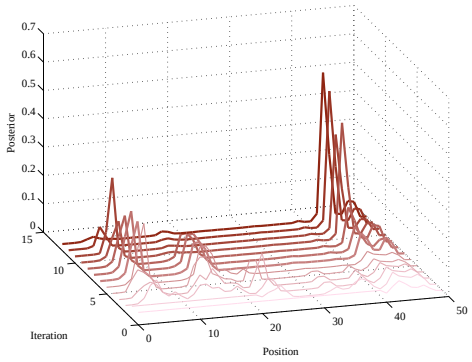


Fig. 10. Depth estimation using Bayesian decision process: evolution of the posterior probability density function over the time. The further graph allows to estimate the depth of the object on the view line. The 3D segment is $1,5m$ long has been sampled in 50 bins. The object is thought to be at position 43. The estimated depth is $1.37m$.

The proposed method will be deployed on the Manus robotic arm to give handicapped people an initialization step for the grasping task. It will be tested and validated in a house environment using everyday life objects. The next step of this work will consist of finding a way to pick up the desired object from the scene bearing in mind that the grasping task can be made easier by decoupling the commands. Eye-in-hand/Eye-to-hand cooperation can be a good way to deal with obstacle avoidance.

## References


[1] R. Basri, E. Rivlin, I. Shishoni. Visual homing: Surfing on the epipoles. *IJCV*, 33(2):117–137, February 1999.

[2] M. Becker et al. GripSee: A gesture-controlled robot for object perception and manipulation. *Autonomous Robots*, 6(2):203–221, 1999.

[3] A.J. Davison, W. Mayol, D. Murray. Real-time localisation and mapping with wearable active vision. *ACM/IEEE ISMAR'03*, pp. 18–27, Tokyo, Oct 2003.

[4] R. O. Duda, P.E. Hart, Stork D. G. *Pattern Classification, second edition*. Wiley Interscience Publication, 2001.

[5] M. Elena, M Critiano, F. Damiano, M. Bonfe. Variable sturcture PID controler for cooperative eye-in-hand/eye-to-hand visual servoing. *IEEE. Int. Conf. on Control Applications, ICCA'03*, pages 989–994, Istambul, 2003.

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

[7] G. Flandin, F. Chaumette, E. Marchand. Eye-in-hand / eye-to-hand cooperation for visual servoing. *IEEE Int. Conf. on Robotics and Automation*, pp. 2741–2746, San Francisco, April 2000.

[8] R. Hartley, A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2001.

[9] R. Horaud, D. Knossow, M. Michaelis. Camera cooperation for achieving visual attention. *Machine Vision and App.*, 16(6), 2006.

[10] V. Lippiello, B. Siciliano, L. Villani. Eye-in-hand/eye-to-hand multi-camera visual servoing. *CDC'05*, pp. 5354-5359, Seville, dec 2005.

[11] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[12] J. Piazzi, D. Prattichizzo, N. J. Cowan. Auto epipolar visual servoing. *IEEE IROS'04*, pp. 363–368, Sendai, oct 2004.

[13] P. Rives. Visual servoing based on epipolar geometry. *IEEE IROS'00*, volume 1, pp. 602–607, Takamatsu, Japan, November 2000.