

Multi-sensor data fusion in sensor-based control: application to multi-camera visual servoing

Olivier Kermorgant, F. Chaumette

▶ **To cite this version:**

Olivier Kermorgant, F. Chaumette. Multi-sensor data fusion in sensor-based control: application to multi-camera visual servoing. IEEE Int. Conf. on Robotics and Automation, ICRA'11, 2011, Shanghai, China, China. pp.4518-4523. hal-00639682

HAL Id: hal-00639682
https://inria.hal.science/hal-00639682

Submitted on 9 Nov 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multi-sensor data fusion in sensor-based control: application to multi-camera visual servoing

Olivier Kermorgant and François Chaumette

***Abstract*— A low-level sensor fusion scheme is presented for the positioning of a multi-sensor robot. This non-hierarchical framework can be used for robot arms or other velocity-controlled robots, and is part of the task function approach. A stability analysis is presented for the general case, then several control laws illustrate the versatility of the framework. This approach is applied to the multi-camera eye-in-hand/eye-to-hand configuration in visual servoing. Experimental results point out the feasibility and the effectiveness of the proposed control laws. Mono-camera and multi-camera schemes are compared, showing that the proposed sensor fusion scheme improves the behavior of a robot arm.**

***Index Terms*— sensor fusion, multi-sensor, visual servoing, multi-camera**

I. Introduction

Most actuated systems use sensors to obtain information about their environment. They can be a camera, ranging devices or temperature or force sensors. The concept of data fusion is the answer to how to combine features coming from the same environment, yet through different sensors.

A wide classification of sensor data fusion approaches is done in [10]. The most popular approach is to use the sensor data to estimate the robot state [4]. For instance, in [6], inertial measurements are used to correct the force and wrist estimation of a force sensor. In [9], several cameras are used together to estimate the 3D-position of an object. In this approach, sensor data are not directly used in the control loop and are part of the real-time estimation of the robot state. A model-based control law can then be applied.

The main alternative to state estimation is named sensor-based control. Here, one does not use the sensors to estimate the system state, but rather sticks to the sensor space. In this approach, each sensor is given a reference signal and considered as an independent subtask of the global task function. A classical scheme, often named hybrid control, is to draw a hierarchy between the different sensors and to build a control scheme that prevents lower subtasks to disturb higher ones [7]. This hierarchy can be made dynamic to prevent from reaching local minima [12]. With another formulation, sensor-based control laws can be designed without imposing a strict hierarchy between the sensors. Here the data coming from different sensors is treated as a unique, higher-dimensional signal. This is the approach chosen in [11] to fuse two cameras, and a force sensor and a camera, where the designed control law is equivalent to a weighted sum of the subtask control laws.

Olivier Kermorgant and François Chaumette are with INRIA Rennes-Bretagne Atlantique, Campus de Beaulieu, Rennes, France.
E-mail: Firstname.Name@inria.fr

Our work is part of the non-hierarchical class of sensor data fusion schemes. Contrary to the previously presented approaches, there is no concept of priority between the different sensors: therefore only the global error is taken into account. The main contribution is the exhibition of a class of easy-tuned control laws that do not require any a priori hierarchy between the sensors and show nice properties in the sensor space and in the robot behavior.

Section II is dedicated to the modeling of multi-sensor-based control. Afterwards, a stability analysis is performed, from which several control laws are proposed. Finally, Section IV illustrates the control laws, with experiments in the case of multi-camera eye-in-hand/eye-to-hand cooperation. Results validate that non-hierarchical schemes can induce nice behaviors for a multi-sensor system.

II. Multi-sensor modeling

This section covers and adapts the modeling used in [11] for the control from multiple sensors. We consider a robotic system equipped with k sensors providing data about the robot pose in its environment. Each sensor S_i delivers a signal $\mathbf{s}_i$ of dimension n_i with $\sum_{i=1}^{k} n_i = n$ and we assume $n \geq 6$. In the case of a motionless environment, the signal time derivative is directly related to the sensor velocity screw $\mathbf{v}_i$ expressed in the sensor frame:

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

where $\mathbf{L}_i$ is named the interaction matrix of $\mathbf{s}_i$ [2], [14] and is of dimension $(n_i \times 6)$. Its analytical form can be derived for many features coming from exteroceptive sensors. It depends mainly on the type of considered sensory data $\mathbf{s}$ and on the sensor intrinsic parameters. $\mathbf{L}_i$ may also depend on other data: for instance the interaction matrix of an image point observed by a camera depends on the depth of that point, which is not actually measured in the image.

Now, we consider a reference frame $\mathcal{F}_e$ in which the robot velocity can be controlled. This frame can be the end-effector frame for a manipulator, or the vehicle body frame for a mobile robot. The screw transformation matrix allows expressing the sensor velocity $\mathbf{v}_i$ wrt. the robot velocity $\mathbf{v}_e$:

$$\mathbf{v}_i = {}^i\mathbf{W}_e \mathbf{v}_e \tag{2}$$

${}^i\mathbf{W}_e$ is given by:

$${}^i\mathbf{W}_e = \left[\begin{array}{c|c} {}^i\mathbf{R}_e & \left[{}^i\mathbf{t}_e\right]_\times {}^i\mathbf{R}_e \\ \hline 0_{3\times 3} & {}^i\mathbf{R}_e \end{array}\right] \tag{3}$$

where ${}^i\mathbf{R}_e \in SO(3)$ and ${}^i\mathbf{t}_e \in \mathbb{R}^3$ are respectively the rotation and the translation between $\mathcal{F}_e$ and $\mathcal{F}_{\mathbf{s}i}$. $\left[{}^i\mathbf{t}_e\right]_\times$ is the (3×3) skew-symmetric matrix related to ${}^i\mathbf{t}_e$. From (1) and (2) we

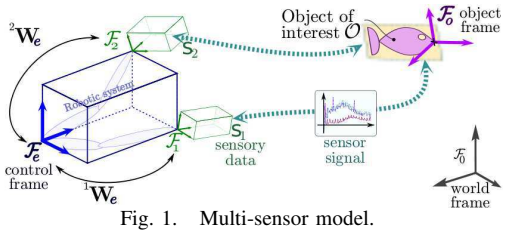


Fig. 1. Multi-sensor model.

can express the time variation of a sensor signal wrt. the robot velocity screw:

$$\dot{\mathbf{s}}_i = \mathbf{L}_i {}^i\mathbf{W}_e \mathbf{v}_e \tag{4}$$

Denoting $\mathbf{s} = (\mathbf{s}_1, \dots, \mathbf{s}_k)$ the n-dimensional signal of the multi-sensor set, (4) allows linking the signal time variation with the robot velocity:

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v}_e \tag{5}$$

with:

$$\mathbf{L}_s = \begin{bmatrix} \mathbf{L}_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \mathbf{L}_k \end{bmatrix} \begin{bmatrix} {}^1\mathbf{W}_e \\ \vdots \\ {}^k\mathbf{W}_e \end{bmatrix} = \mathbf{L}\mathbf{W}_e \tag{6}$$

where $\mathbf{L} \in \mathbb{R}^{n\times 6k}$ contains the interaction matrices of the sensors and $\mathbf{W}_e \in \mathbb{R}^{6k\times 6}$ contains the transformation matrices, making $\mathbf{L}_s \in \mathbb{R}^{n\times 6}$ the global interaction matrix of the task. Note that this modeling can be expressed in any frame used for the robot control: indeed, for any frame $\mathcal{F}_f$ we have $\mathbf{W}_e\mathbf{v}_e = \mathbf{W}_e {}^e\mathbf{W}_f \mathbf{v}_f = \mathbf{W}_f \mathbf{v}_f$.

The goal of multi-sensor servoing is to design a control law that makes the robot reach the desired sensor measurement $\mathbf{s}^*$. To do so, a simple proportional control law yields:

$$\mathbf{v}_e = -\lambda \mathbf{C}(\mathbf{s} - \mathbf{s}^*) \tag{7}$$

where λ is a positive scalar gain and $\mathbf{C} \in \mathbb{R}^{6\times n}$ is named the combination matrix and has to be defined. For instance, if we want to ensure an exponential decoupled decrease of the error $(\mathbf{s} - \mathbf{s}^*)$, then (5) leads to:

$$\mathbf{C} = \mathbf{L}_s^+ \tag{8}$$

$\mathbf{L}_s^+$ being the Moore-Penrose pseudo-inverse of $\mathbf{L}_s$, that is $\mathbf{L}_s^+ = (\mathbf{L}_s^\intercal \mathbf{L}_s)^{-1}\mathbf{L}_s^\intercal$ when $\mathbf{L}_s$ is of full rank 6. In practice, an estimation $\widehat{\mathbf{L}_s^+}$ is used, as both $(\mathbf{L}_i)_i$ and $\left({}^i\mathbf{W}_e\right)_i$ depend on potentially unknown parameters and are subject to noise or calibration errors: $\mathbf{L}_i$ relies on the sensor model and the direct sensor measurements, while ${}^i\mathbf{W}_e$ depends on the pose between the sensor and the robot. Actually, the frame transformation matrix ${}^i\mathbf{W}_e$ is subject to change during the task when the sensor S_i is not rigidly attached to the robot control frame, which is the case for instance for an eye-to-hand system. The final control law thus yields:

$$\mathbf{v}_e = -\lambda \widehat{\mathbf{L}_s^+}(\mathbf{s} - \mathbf{s}^*) \tag{9}$$

where:

$$\widehat{\mathbf{L}_s^+} = (\hat{\mathbf{L}}\hat{\mathbf{W}}_e)^+ \tag{10}$$

The system behavior thus depends on the chosen estimation of $\mathbf{L}$ and $\mathbf{W}_e$. The pseudo-inverse inherits the anti-commutativity from the classical inverse, that is $(\hat{\mathbf{L}}\hat{\mathbf{W}}_e)^+ = \hat{\mathbf{W}}_e{}^+\hat{\mathbf{L}}^+$, when $\hat{\mathbf{L}}$ (resp. $\hat{\mathbf{W}}_e$) has orthonormal columns (resp. rows) or when both matrices are of full rank equal to their common dimension [1]. For the proposed configuration, this property is ensured in two cases:

1) $\hat{\mathbf{W}}_e$ has orthonormal rows, which is equivalent to having only one sensor and no translation between this sensor and the control frame (see (3))
2) $\hat{\mathbf{L}}$ and $\hat{\mathbf{W}}_e$ are both of rank $6k$, which can be ensured again if there is only one sensor, using $n \geq 6$ features that induce a full-rank interaction matrix

Therefore as soon as one fuses several sensors, (10) must be used as such.

III. Stability analysis

From (5) and (9), the evolution of the sensor signal yields:

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v}_e = -\lambda \mathbf{L}_s \mathbf{C}(\mathbf{s} - \mathbf{s}^*) \tag{11}$$

The proportional scheme (11) has different stability properties depending on the number of features [2].

a) $n=6$: denoting the signal error $\mathbf{e} = (\mathbf{s} - \mathbf{s}^*)$, we can define the candidate Lyapunov function that is the squared error norm $\mathcal{L} = \frac{1}{2}\|\mathbf{e}\|^2$, whose deritative yields:

$$\dot{\mathcal{L}} = \mathbf{e}^\intercal \dot{\mathbf{e}} = -\lambda \mathbf{e}^\intercal \mathbf{L}_s \mathbf{C} \mathbf{e} \tag{12}$$

Hence, using the combination matrix defined in (8) the system is globally asymptotically stable (GAS) if the following condition is ensured:

$$\mathbf{L}_s\mathbf{C} = \mathbf{L}_s\widehat{\mathbf{L}_s^+} = \mathbf{L}\mathbf{W}_e(\hat{\mathbf{L}}\hat{\mathbf{W}}_e)^+ > 0 \tag{13}$$

Therefore, if the features and sensor positions are chosen so that $\mathbf{L}_s$ and $\widehat{\mathbf{L}_s^+}$ are of full rank, the system is stable as soon as the estimations of $\mathbf{L}$ and $\mathbf{W}_e$ are close enough from their actual value, the ideal case being of course $\hat{\mathbf{L}}\hat{\mathbf{W}}_e = \mathbf{L}\mathbf{W}_e$.

b) $n > 6$: here condition (13) is never ensured as $\mathbf{L}_s\mathbf{C} \in \mathbb{R}^{n\times n}$ is of rank 6 at most. This means local minima may exist, for configurations such that $\mathbf{e} \in \mathrm{Ker}\mathbf{C}$. Stability can thus be at most local. Defining the task error as $\mathbf{e}' = \mathbf{C}\mathbf{e}$ yields:

$$\begin{aligned} \dot{\mathbf{e}}' &= \mathbf{C}\dot{\mathbf{e}} + \dot{\mathbf{C}}\mathbf{e} = (\mathbf{C}\mathbf{L}_s + \mathbf{O})\mathbf{v}_e \\ &= -\lambda(\mathbf{C}\mathbf{L}_s + \mathbf{O})\mathbf{e}' \end{aligned} \tag{14}$$

where $\mathbf{O} \in \mathbb{R}^{6\times 6} = 0$ when $\mathbf{e} = 0$ [2]. With the combination matrix from (8), this scheme is known to be locally asymptotically stable in a neighborhood of $\mathbf{e} = \mathbf{e}^* = 0$ [8] if:

$$\mathbf{C}\mathbf{L}_s = \widehat{\mathbf{L}_s^+}\mathbf{L}_s = (\hat{\mathbf{L}}\hat{\mathbf{W}}_e)^+\mathbf{L}\mathbf{W}_e > 0 \tag{15}$$

If the features and sensor positions are chosen so that $\mathbf{L}_s$ and $\widehat{\mathbf{L}_s^+}$ are of full rank 6, the system is locally stable as soon as the estimations of $\mathbf{L}$ and $\mathbf{W}_e$ are not too coarse.

Finally, when the number of features is sufficient, that is $n \geq 6$, the stability mainly depends on the chosen estimation for $\widehat{\mathbf{L}_s^+}$. For this purpose, several control laws are proposed. The final choice depends of which information is available during the servoing:

1) $\mathbf{C}_{\text{cur}} = (\mathbf{L}\mathbf{W}_e)^+$ when both matrices can be measured (usually with noise) during the task.

2) $\mathbf{C}_* = (\mathbf{L}^*\mathbf{W}_e)^+$ if $\mathbf{W}_e$ is measured, while $\mathbf{L}^*$ is computed at the desired position.
3) $\mathbf{C}_3 = (\mathbf{L}^*\mathbf{W}_e^*)^+$ uses the values computed at the desired position. In this case, the combination matrix is constant. This choice is equal to the previous one when $\mathbf{W}_e$ is constant, that is for instance for an eye-in-hand system.
4) $\mathbf{C}_4 = (\overline{\mathbf{L}}\mathbf{W}_e)^+$ where $\overline{\mathbf{L}}$ is the mean interaction matrix allowing second-order minimization [15], that is
$$\overline{\mathbf{L}} = \begin{bmatrix} \frac{1}{2}(\mathbf{L}_1 + \mathbf{L}_1^{*1^*}\mathbf{W}_1) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \frac{1}{2}(\mathbf{L}_k + \mathbf{L}_k^{*k^*}\mathbf{W}_k) \end{bmatrix}.$$
5) $\mathbf{C}_5 = (\frac{1}{2}(\mathbf{L}\mathbf{W}_e + \mathbf{L}^*\mathbf{W}_e^*)^+$ is known to have nice properties in the visual servoing case [2] and is an approximation of the previous one.
6) $\mathbf{C}_{\frac{1}{2}} = (\frac{1}{2}(\mathbf{L} + \mathbf{L}^*)\mathbf{W}_e)^+$ is another approximation of the second order minimization, by setting ${}^{i^*}\mathbf{W}_i = \mathbb{I}_6$. Again, this choice is equal to the previous one when $\mathbf{W}_e$ is constant.

When a sensor is not rigidly attached to the effector, for instance in an eye-to-hand case, $\mathbf{W}_e$ is not constant and the desired value $\mathbf{W}_e^*$ depends on the final 3D pose of the sensors wrt. the effector. This pose is generally unknown, making schemes $\mathbf{C}_3$ and $\mathbf{C}_5$ implausible choices. $\mathbf{C}_4$ involves the knowledge of the pose error for each sensor ${}^{i^*}\mathbf{W}_i$, which is usually not known either. $\mathbf{C}_{\frac{1}{2}}$ is thus an acceptable estimation for second order minimization. That is why only $\mathbf{C}_{\text{cur}}$,$\mathbf{C}_*$ and $\mathbf{C}_{\frac{1}{2}}$ are considered in the next section. When reaching the final position all possibilities lead to the same behavior, hence local stability, as all combination matrices converge towards $\mathbf{C}_*$. However, the behaviors can be very different when $\mathbf{s}$ is far from $\mathbf{s}^*$.

Additionally, in [11] the authors designed a combination matrix such that the resulting control law is a weighted sum of each subsystem task function (hence named subsystem-based fusion):

$$\mathbf{C} = \mathbf{C}_\# = \begin{bmatrix} \kappa_1 {}^1\mathbf{W}_e^{-1}\mathbf{L}_1^+ & \dots & \kappa_k {}^k\mathbf{W}_e^{-1}\mathbf{L}_k^+ \end{bmatrix} \quad (16)$$

leading to the following control law:

$$\mathbf{v}_e = -\lambda \sum_{i=1}^{k} \kappa_i {}^i\mathbf{W}_e^{-1}\mathbf{L}_i^+(\mathbf{s}_i - \mathbf{s}_i^*) = \sum_{i=1}^{k} \kappa_i \mathbf{v}_i \quad (17)$$

with $\kappa_i > 0$ and $\sum_{i=1}^{k} \kappa_i = 1$. This formulation is intuitive when each subsystem is of rank 6, and allows a very easy stability analysis: indeed, a sufficient condition for the whole system to be stable is that each subsystem is stable, the main drawback being that the non-stability of one subsystem prevents from ascertaining the stability of the whole system. However, not taking into account the coupling of the subsystems prevents from ensuring a decoupled decrease of the error near the convergence. This approach also makes it necessary to tune every subsystem gain in order to obtain a generic acceptable behavior: this is mainly due to the usually unbalanced velocity contribution from each sensor. In the next section, the combination matrix $\mathbf{C}_\#$ is also compared in order to point out these unbalanced velocity contributions and the non-decoupling of the convergence.

Finally, the behavior is of course highly depending on which features are used by each sensor and the sensor poses wrt. the robot frame, that define the structure of $\mathbf{L}$ and $\mathbf{W}_e$.

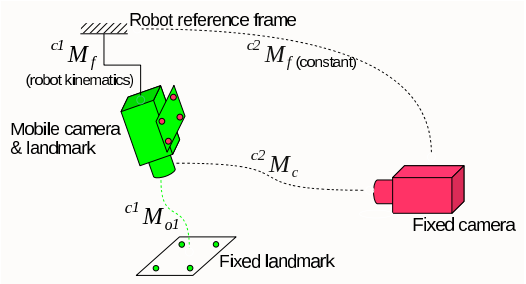


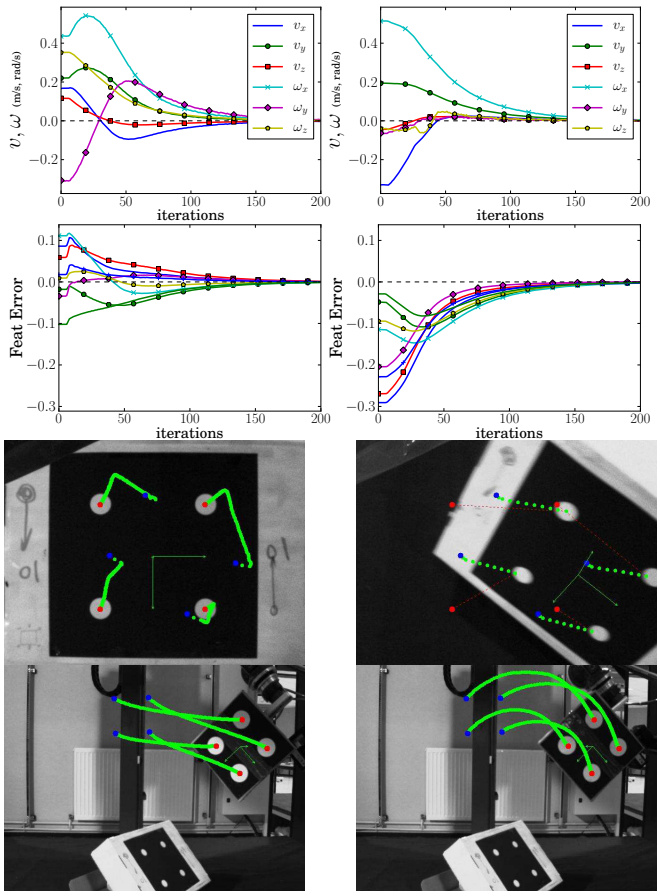
Fig. 2. Eye-in-hand/eye-to-hand cameras. Pose between the two cameras is computed from robot kinematics and initial calibration. Pose between a camera and its landmark is computed during the task

IV. Experimental results

Experiments have been carried on a 6 degrees-of-freedom Gantry robot, the control laws are implemented using ViSP software [13]. We have chosen to perform a visual servoing (VS) image-based task [2] with two cameras. VS configurations can be divided into two categories: eye-in-hand, when the camera is attached to the end-effector, and eye-to-hand when the camera is fixed in the workspace. The combination of the two is called eye-in-hand/eye-to-hand cooperation and is a classical VS approach that leads to many applications [9], [3], [5]. This configuration allows a versatile comparison of the proposed sensor data fusion schemes: indeed, the two sensors are not rigidly attached, and known or unknown current interaction matrix represent common situations in visual servoing. We first describe the experimental conditions before presenting the behavior of the robot controlled by only one camera at a time. Afterwards, several multi-camera schemes are compared.

A. Experimental conditions

The eye-in-hand camera $\mathcal{F}_{c1}$ is observing a fixed landmark $\mathcal{F}_{o1}$ composed of four points. The end-effector carries a second landmark $\mathcal{F}_{o2}$ also composed of four points, which is observed by the eye-to-hand camera $\mathcal{F}_{c2}$. $\mathbf{s}$ is composed of 2D-point cartesian coordinates, hence both subsystems are of dimension $n_i = 8$, making $\mathbf{s}$ of dimension $n = 16$. A pose computation is performed from each image thanks to the knowledge of the landmark dimensions, to make it possible to estimate the depth of each point and thus use the current interaction matrix if needed. The objective of the task is to position the end-effector so that both cameras reach their respective reference image. To do so, the robot is controlled in the eye-in-hand camera frame where all velocities are expressed. As previously said in Section II, any other control frame would lead to the same results.
To make it possible to express all velocities in $\mathcal{F}_{c1}$, the pose between the fixed camera and the fixed robot frame, that is ${}^{c2}\mathbf{M}_f$, is calibrated. From a configuration making both cameras observe the fixed landmark, pose computation is performed from each image that gives ${}^{c1}\mathbf{M}_{o1}$ and ${}^{c2}\mathbf{M}_{o1}$. We



a) Eye-in-hand scheme b) Eye-to-hand scheme

Fig. 3. Mono-camera, eye-in-hand (left) and eye-to-hand (right) VS. Velocities, feature errors and trajectories in eye-in-hand (top) and eye-to-hand (bottom) images.

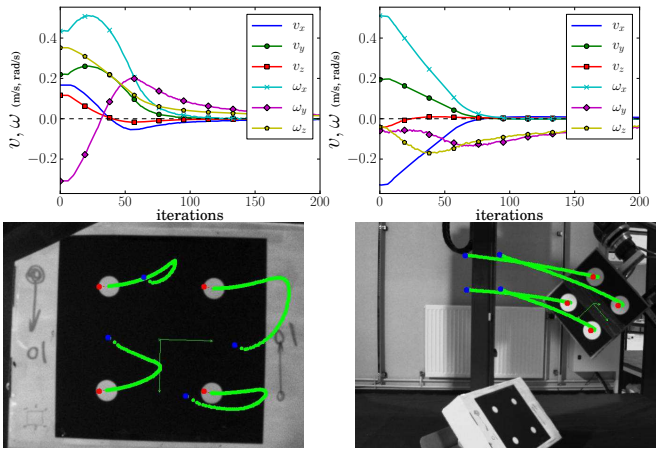
deduce ${}^{c2}\mathbf{M}_f = {}^{c2}\mathbf{M}_{o1}\,{}^{c1}\mathbf{M}_{o1}^{-1}\,{}^{c1}\mathbf{M}_f$ where ${}^{c1}\mathbf{M}_f$ is available through the robot geometrical model and odometry.

In order to avoid any incompatibility between the two reference images, we initially choose a desired pose $(0, 0, 0.5, 0, 0, 0)$ (m,deg) for the eye-in-hand camera, which means we want to have the camera 0.5 m from the landmark, the latter drawing a centered square in the image. From this pose, a reference image is computed and a single, eye-in-hand camera VS task is performed. When the convergence is obtained, the corresponding eye-to-hand image is acquired and is then used as the desired eye-to-hand image in the multi-camera schemes.

The initial eye-in-hand pose is $(0.02, 0.02, 0.52, 47, 12, 28)$. An ideal behavior would be to follow a geodesic in 3D space, while having trajectories in the image without approaching the image borders.

B. Mono-camera behavior

The first mono-camera VS scheme is the eye-in-hand configuration with the desired interaction matrix, represented in Fig. 3a. Actually, even if only the eye-in-hand camera is used, the eye-to-hand features have a nice trajectory in the image. On a second step, we perform an eye-to-hand VS with the desired interaction matrix, the results of which are shown in Fig. 3b. This time, some of the eye-in-hand features quickly leave the image range, while the task is performed in the eye-to-hand image. A case of unbalanced velocity contribution can be seen on the velocity figures: indeed, v_x is the opposite from one scheme to another. That is why the eye-in-hand points have a totally irrelevant x-motion in the eye-to-hand scheme. Hence, mono-camera schemes do not have the same behavior although they are built on coherent features. Of course, fusing several sensors does not systematically lead to unbalanced velocity contribution but this case allows illustrating the proposed fusion schemes.



a) Eye-in-hand camera b) Eye-to-hand camera

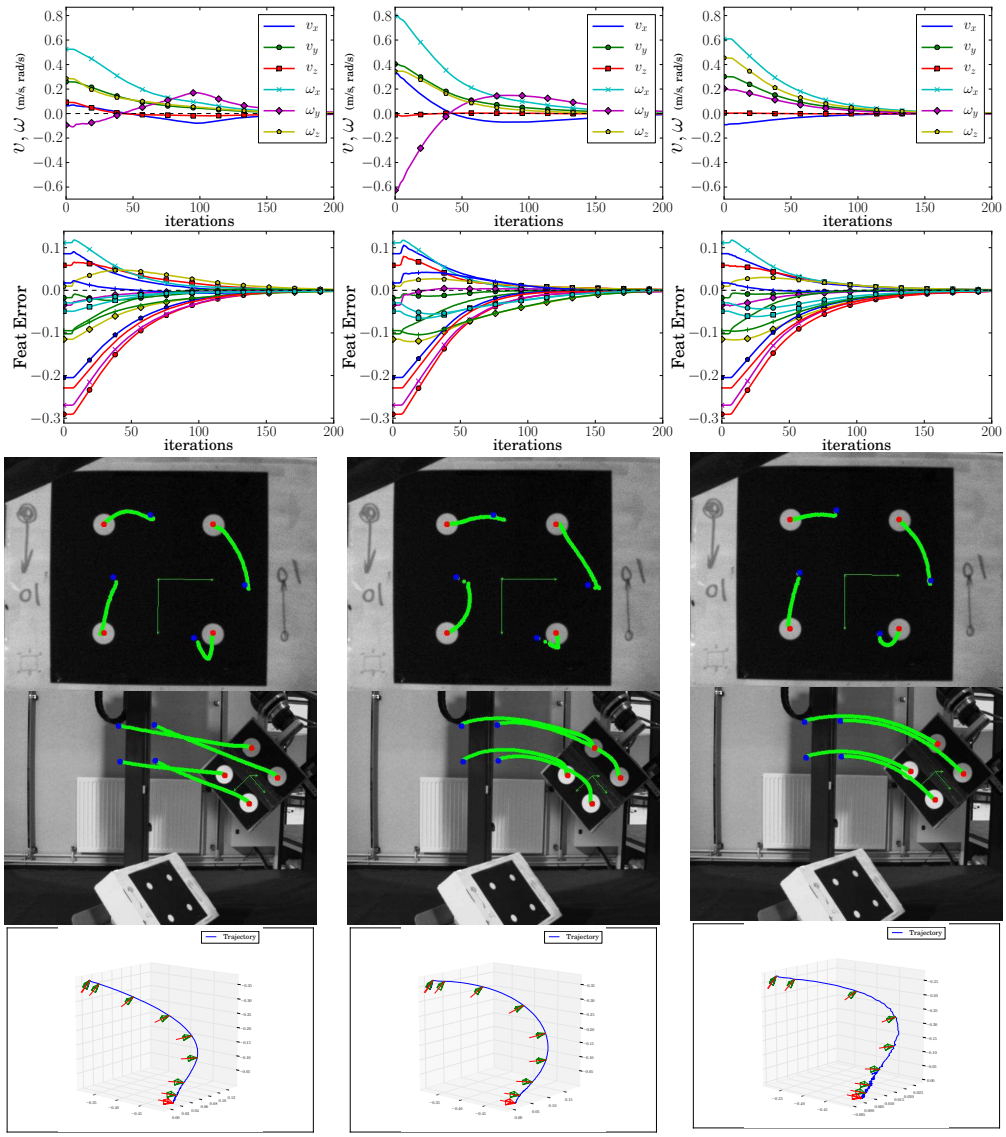
Fig. 4. Subsystem-based fusion $\mathbf{C}_{\#}$. Unweighted velocity contributions and trajectories for eye-in-hand (left) and eye-to-hand (right) cameras. The eye-to-hand contribution is more important, the eye-in-hand camera nearly loses some features

C. Multi-camera behavior

At first, the subsystem-based fusion $\mathbf{C} = \mathbf{C}_{\#}$ is performed. If equal weights are chosen in (17), that is $\kappa_1 = \kappa_2 = 0.5$, the behavior induced by the eye-to-hand view is predominant and the system acts like in the mono-camera, eye-to-hand scheme: eye-in-hand features go out of the image range. This can be explained by the different conditionning of the interaction matrices: indeed, for a point the interaction matrix depends on the inverse depth, and here the desired depth is 0.5m for the eye-in-hand view and about 1.2m for the eye-to-hand view. Therefore, the scheme has to be tuned and we choose a lower weight for the eye-to-hand subsystem: $\kappa_1 = 0.7$ and $\kappa_2 = 0.3$. The results of the weighted task are shown in Fig. 4, where the eye-in-hand features are still close from leaving the image.

On the contrary, the three proposed low-level fusion schemes are performed without any tuning. Results are given in Fig. 5. The use of $\mathbf{C}_{\text{cur}}$ leads to a very nice behavior in both images, but the robot trajectory is not satisfactory. As for the desired matrices configuration, results show that the eye-in-hand behavior is improved compared to IV-B, by taking into account the eye-to-hand view. An even better behavior is obtained by using the mean value $\mathbf{C}_{\frac{1}{2}}$. Results show very regular velocities and error decrease, while trajectories in both images are quite smooth. Actually, this scheme also induces a smooth 3D trajectory of the end-effector.

First experiments show the robustness of low-level fusion



a) $\mathbf{C}_{\text{cur}}$ b) $\mathbf{C}_*$ c) $\mathbf{C}_{\frac{1}{2}}$

Fig. 5. Fusion schemes: Velocities (top), feature error (middle), eye-in-hand and eye-to-hand images, and 3D trajectory of $c1$. All schemes converge without any subtask weighting. Image trajectories are satisfactory, but 3D behavior is far from geodesic for $\mathbf{C}_{\text{cur}}$.

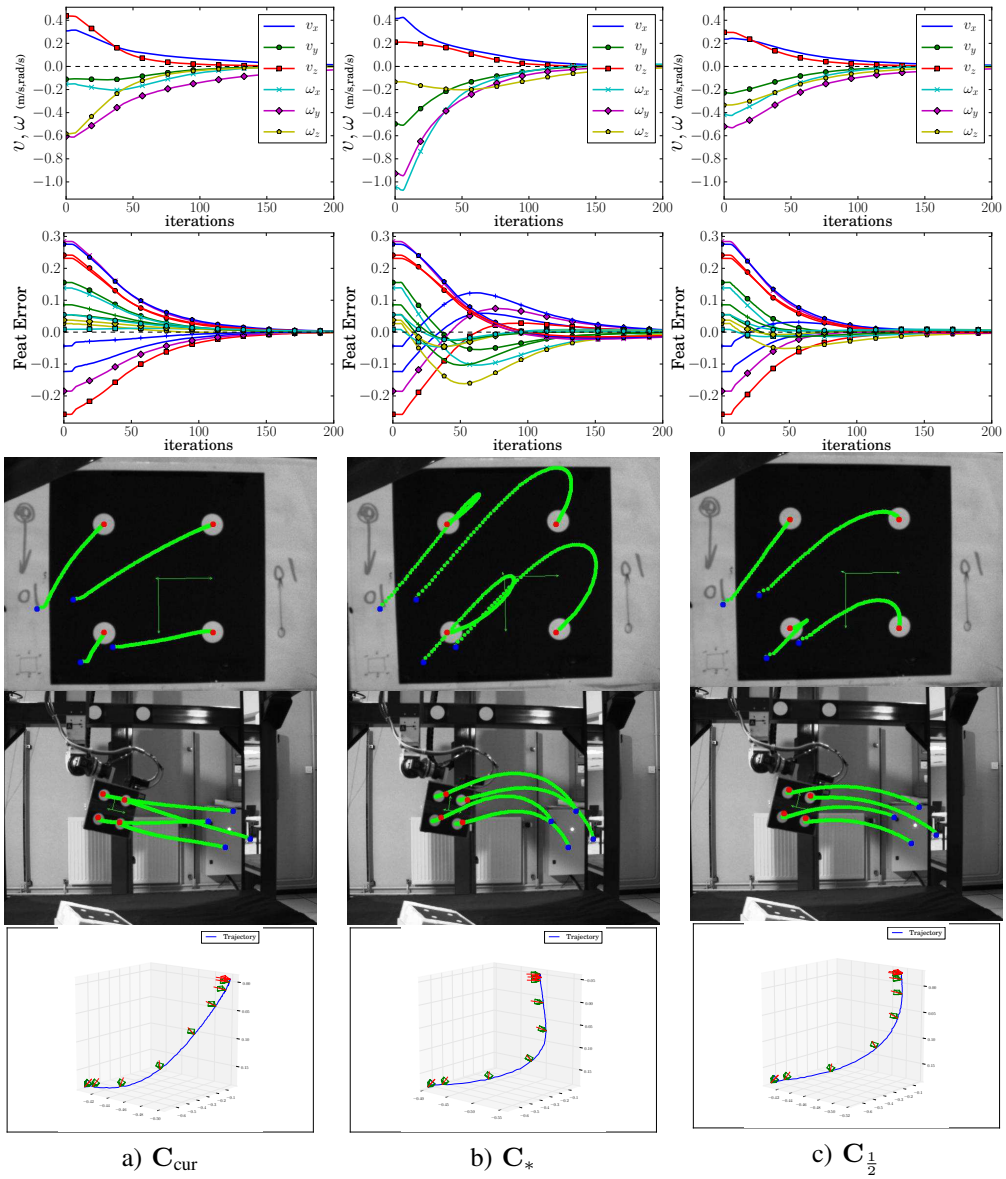
when using the different proposed schemes. In order to illustrate the possible limitations, new runs are performed with another initial position $(-0.12, 0.08, 0.8, -33, -63, -30)$, necessiting more complex motion and increasing the desired eye-to-hand depth to 1.4 m.

Mono-camera behaviors (not represented) still indicate that with the new initial position, eye-in-hand scheme performs as in Fig. 3a and eye-to-hand scheme as in Fig. 3b, making eye-in-hand features leave the image range. This is confirmed by the subsystem-based fusion scheme (not represented) that can be achieved again only if the subtask weights are adequately tuned.

As for the proposed low-level fusion schemes, this time they do not behave the same. Schemes using $\mathbf{C}_{\text{cur}}$ or $\mathbf{C}_{\frac{1}{2}}$ still converge to the desired position with nice behaviors in both images (Fig. 6a,c). Both schemes induce also satisfactory 3D-trajectories. On the opposite, the use of $\mathbf{C}_*$ suffers from the large distance from the initial pose to the desired one, as eye-in-hand trajectory is not satisfactory (Fig. 6b).

V. Conclusions

By using the global sensor signal, a multi-sensor fusion scheme can be designed without imposing any hierarchy between the sensors. The stability analysis concurs to the classical conditions on the system rank, with potential local minima in the case of redundancy. Several control laws have been compared in order to illustrate the proposed generic scheme, in the case of image-based visual servoing. Although unsatisfactory cases may be encountered for some control laws when the displacement to realize is very large, experiments highlight the advantages of these schemes and the nice behavior they induce wrt. mono-camera schemes or a weighted multi-camera control law. This framework does not depend on the sensor nature, making possible to extend the proposed scheme to other types of sensors or visual features.



a) $\mathbf{C}_{\text{cur}}$ b) $\mathbf{C}_*$ c) $\mathbf{C}_{\frac{1}{2}}$

Fig. 6. Fusion schemes for new initial position. Nice behaviors are found with current (a) and mean (c) combination matrix. $\mathbf{C}_*$ (b) draws unsatisfactory image behavior, making eye-in-hand view nearly lose some features.

ACKNOWLEDGMENT

The authors acknowledge support from the ANR Psirob SCUAV project of the French ministry of research.

REFERENCES

[1] A. Albert, *Regression and the Moore-Penrose pseudoinverse*. Academic Pr, 1972.

[2] F. Chaumette and S. Hutchinson, "Visual servo control, part I: Basic approaches," *IEEE Robot. Autom. Mag.*, vol. 13, no. 4, Dec. 2006.

[3] C. Dune, E. Marchand, and C. Leroux, "One click focus with eye-in-hand/eye-to-hand cooperation," in *ICRA'07*, Roma, Italy, April 2007.

[4] H. Durrant-Whyte, *Integration, Coordination and Control of Multi-Sensor Robot Systems*. Kluwer Academic Publishers, 1987.

[5] G. Flandin, F. Chaumette, and E. Marchand, "Eye-in-hand / eye-to-hand cooperation for visual servoing," in *ICRA'00*, San Francisco, California, April 2000.

[6] J. Gamez Garcia, A. Robertsson, J. Gomez Orteg, and R. Johansson, "Sensor Fusion for Compliant Robot Motion Control," *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 430–441, 2008.

[7] K. Hosoda, K. Igarashi, and M. Asada, "Adaptive hybrid visual servoing/force control in unknown environment," in *Proc. IROS*, 1996.

[8] A. Isidori, *Nonlinear control systems*. Springer Verlag, 1995.

[9] V. Lippiello, B. Siciliano, and L. Villani, "Position-based visual servoing in industrial multirobot cells using a hybrid camera configuration," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 73–86, 2007.

[10] R. Luo, C. Yih, and K. Su, "Multisensor fusion and integration: approaches, applications, and future research directions," *IEEE Sensors Journal*, vol. 2, no. 2, pp. 107–119, 2002.

[11] E. Malis, G. Morel, and F. Chaumette, "Robot control using disparate multiple sensors," *Int. J. of Robotics Research*, vol. 20, no. 5, pp. 364–377, May 2001.

[12] N. Mansard and F. Chaumette, "Task sequencing for sensor-based control," *IEEE Trans. Robot.*, vol. 23, no. 1, pp. 60–72, Feb. 2007.

[13] E. Marchand, F. Spindler, and F. Chaumette, "Visp for visual servoing: a generic software platform with a wide class of robot control skills," *IEEE Robot. Autom. Mag.*, vol. 12, no. 4, December 2005.

[14] C. Samson, M. L. Borgne, and B. Espiau, *Robot Control : The task function approach*. Clarendon Press, 1991.

[15] O. Tahri and Y. Mezouar, "On visual servoing based on efficient second order minimization," *Robotics and Autonomous Systems*, 2009.