

Kinematic Calibration of a Gough-Stewart Platform Using an Omnidirectional Camera

Tej Dallej, Hicham Hadj-Abdelkader, Nicolas Andreff and Philippe Martinet
LASMEA - CNRS - Université Blaise Pascal/IFMA, 63175 Aubière, France
Email: {firstname.lastname}@lasmea.univ-bpclermont.fr
<http://www.lasmea.univ-bpclermont.fr/Control>

Abstract—This paper is related to the vision-based control of parallel robots. Indeed, a method is proposed to estimate the reduced set of kinematic parameters appearing in such a control. To do so, it extends a linear method, obtained for a perspective camera, to the case of an omnidirectional camera, using an existing unifying projection model. The proposed method remains linear, once adequate information is extracted from the omnidirectional images, and does not require any calibration pattern. It works both with perspective and omnidirectional cameras, which is underlined by the reported experiments.

I. INTRODUCTION

Controlling parallel robots is a hard task. Indeed, simple joint control does not take into account the kinematic constraints due to the closed kinematic chains of such mechanisms. Hence, it may yield high internal forces [1] and can only achieve a good repeatability (relative positioning error). The latter is generally higher than in the serial case [2] and is due mainly to the mechanical constraints, while the achievable accuracy (absolute positioning error) is usually in the same order as serial robot accuracy [3]. The latter property comes from an inaccurate conversion from desired Cartesian pose to desired joint values, mainly due to the fact that the large number of links and passive joints makes it very hard to manufacture and assemble a physical parallel robot close to its CAD model. Therefore, one has to use advanced control techniques, that belong to two classes.

The first class of control is model-based Cartesian control and is extensively described in the serial case. This class of control requires an accurate model of the robot, since the feedback signal is the estimated Cartesian pose of the end-effector from the joint values through the forward kinematic model.

The accuracy of the model is obtained by calibration, which, in the case of parallel robots, can be achieved in several manners. The first one uses the forward kinematic model, which Daney [4] prevents us from. The second one uses mechanical constraints, as in [5] for instance. A third manner uses exteroceptive sensors and either the inverse kinematic model or the implicit kinematic model [6]. Finally, a fourth manner is to add extra sensors in the passive joints [7]. However, huge trouble arises in model-based control since one has to solve the theoretically hard forward kinematic problem. Indeed, this problem rarely has a unique solution [8], [9]. This trouble can be overcome by designing robots with forward

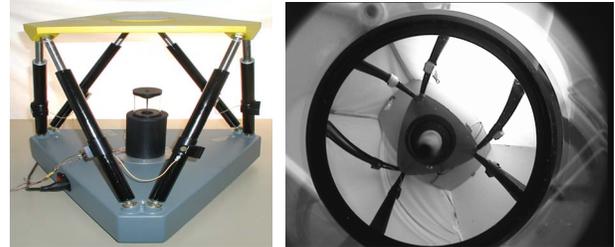


Fig. 1. A Gough-Stewart platform observed by an omnidirectional camera.

kinematic model having a closed-form expression [10] or by adding extra sensors in the passive joints [11].

The latter solution is at the edge of model-based control and sensor-based control, which is the second class of efficient control for parallel robots. In this class, the fundamental assumption is that external sensing of the end-effector pose in the feedback signal replaces advantageously the forward kinematic model, since the perception models are simpler than the kinematic models and contain less unmodelled physical phenomena. Visual servoing [12] is a representative subclass of sensor-based control but was seldom applied to parallel robotics [13], [14], [15].

Sensor-based control relieves us from the forward kinematic problem and partially from the calibration problem. Indeed, if control is performed in the very sensor space (*i.e.* using directly the signal delivered by the sensor), then it only makes use of the model parameters in the so-called interaction matrices [16]. Thus, only a coarse estimation of those parameters is needed (coarse calibration) to ensure convergence. However, the models that are used contain all the kinematic parameters, and most often additional parameters representing the relative pose of the sensor elements (for instance, a camera and a visual pattern, or a laser tracker and a retroreflective cube) with respect to the mechanism.

Alternately, a method was proposed in [17] for vision-based control of a Gough-Stewart platform [18], [19] which has a reduced set of kinematic parameters and does not require any visual target. Indeed, it made use of the fact that the Gough-Stewart platform has generally cylindrical legs. Thus, those cylinders were observed to extract, directly from the image, the leg directions and the latter were used as visual primitives in the control. The advantage of this method, in terms of



Fig. 2. A Gough-Stewart platform observed by a perspective camera.

calibration, is that it does not only reduce the parameter set but also makes calibration a linear problem.

However, the observation of the legs by a single perspective camera causes a self-occlusion problem since the legs in the background of the image may be hidden by those in the foreground (Fig. 2). Moreover, the perspective projection of these legs appear nearly parallel in the image, which is a noise sensitive configuration for the leg direction extraction. A way to solve for the occlusion problem is to turn oneself to a multi-camera perception system but one would then need to calibrate the relative positions of the cameras.

Omnidirectional cameras overcome these problems since they provide a 360 degrees field of view of the surroundings. Many applications in robotics, such as mobile robot localization [20] and navigation [21], can benefit from such a panoramic field of view. Visual servoing applications can also benefit from the cameras with a wide field of view to overcome the visibility constraint. Vision-based control of robot arms [22], [23], single mobile robot [24] or formation of mobile robots [25] appear thus in the literature with omnidirectional cameras. In the literature, there have been several methods proposed to increase the field of view of camera systems [26]. One effective way is to combine mirrors with conventional cameras. The obtained sensors are referred to as catadioptric imaging systems. The resulting imaging systems have been termed central catadioptric when a single projection center describes the world-to-image mapping. The entire class of catadioptric systems with a single view point was derived in [27], while [28] introduced a unifying model for the all central catadioptric imaging systems where the conventional perspective camera appears as a particular case. This model is used in [29] together with the geometry of the catadioptric projection of lines in order to calibrate central catadioptric cameras.

The contribution of this paper is to use this unifying model to extend the calibration method proposed in [17] to the omnidirectional case without losing the linear property. It allows to reconstruct, directly from the images, the geometry of the Gough-Stewart platform base, expressed in the camera frame. If needed, this method could also be applied to determine the geometry of the moving platform, but it will be shown that the vision-based control does not need this information. Moreover, it should be born in mind that this calibration procedure does neither require any mechanical change of the robot nor any calibration pattern, making it very easy to set-up.

The versatility of the method is shown through experimental validation using either an omnidirectional camera observing

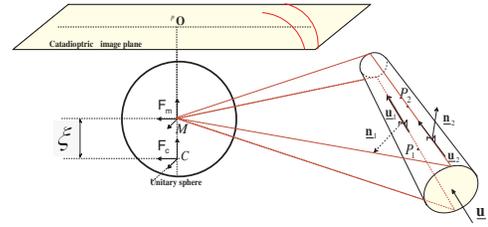


Fig. 3. Projection of a cylinder onto conics in the image.

the legs placed in between them (Fig. 1) or a standard perspective camera observing the legs from the outer side of the mechanism (Fig. 2). In the perspective case, validation is even pushed one step forward by analyzing the convergence of vision-based control by observation of the legs.

Section II recalls the omnidirectional projection model and the derived projection of a cylinder. Section III recalls the vision-based framework for expressing the robot kinematics and control. Then, the calibration method is presented in Section IV and validated in Section V. Conclusions are to be found in Section VI.

II. CATADIOPTIC IMAGE PROJECTION OF A CYLINDER

In this section, we describe the 3D line representation and then we present the catadioptric image formation of 3D lines and cylinders.

A. Line representation

Let \mathcal{L} be a 3D line. A point-independent representation of this line are the Plücker coordinates $(\underline{\mathbf{u}}, \mathbf{n})$ [30] (also known as normalized Plücker coordinates since $\underline{\mathbf{u}}$ is a unit vector), where $\underline{\mathbf{u}}$ is the direction of the line and \mathbf{n} encodes its position.

However, noticing that \mathbf{n} is orthogonal to the so-called interpretation plane defined by \mathcal{L} and the origin, one can split it into two parts: $\underline{\mathbf{n}}$, the unit vector defining the interpretation plane and n , its norm which is the orthogonal distance of \mathcal{L} to the origin.

Doing so, [31] defined the so-called binormalized Plücker coordinates $(\underline{\mathbf{u}}, \underline{\mathbf{n}}, n)$, where only $\underline{\mathbf{n}}$ is useful for catadioptric image projection as we show it below.

B. Unified line projection

Let us recall the central catadioptric projection of 3D lines using the unified model [29] adapted to the representation above. Remind that this model is based on a perspective camera observing the reflection of the scene in a spherical mirror with unitary radius [28].

Let \mathcal{F}_c and \mathcal{F}_m be the frames attached to the camera and to the mirror respectively. Without lost generality, it assumes that these two frames are related by a simple translation by ξ along the z -axis as shown in Fig. 3. In the remainder, the reference frame will be denoted by a left upper-script.

Consider a 3D line \mathcal{L} defined by its binormalized Plücker coordinates expressed in the mirror frame $({}^m\underline{\mathbf{u}}, {}^m\underline{\mathbf{n}}, {}^m n)$. Hence, ${}^m\underline{\mathbf{n}} = (n_x, n_y, n_z)$ is the unit vector expressed in

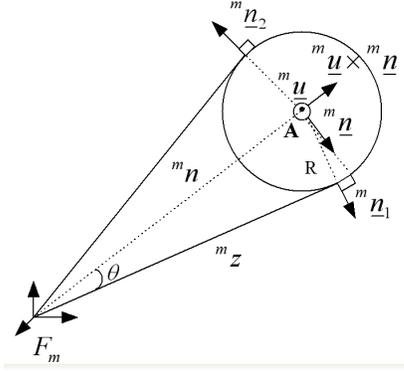


Fig. 4. Visual edges of a cylinder.

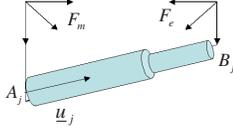


Fig. 5. Parameter setting for each leg.

the mirror frame and orthogonal to the so-called interpretation plane defined by \mathcal{L} and the center \mathbf{M} of the sphere.

According to [23], [28], the projection of \mathcal{L} in the normalized image plane expressed in \mathcal{F}_m is a conic curve defined by the following bilinear form:

$${}^m\Omega \propto \begin{pmatrix} (1-\xi^2)n_x^2 - n_z^2\xi^2 & (1-\xi^2)n_x n_y & n_x n_z \\ (1-\xi^2)n_x n_y & (1-\xi^2)n_y^2 - n_z^2\xi^2 & n_y n_z \\ n_x n_z & n_y n_z & n_z^2 \end{pmatrix} \quad (1)$$

where \propto means *proportional to* and ξ is the intrinsic parameter of the mirror. The expression of this form in pixel coordinates is trivially given by:

$${}^p\Omega = \mathbf{K}^{-T} {}^m\Omega \mathbf{K}^{-1} \quad (2)$$

where \mathbf{K} is the matrix of intrinsic parameters.

Note that the particular case of the conventional perspective camera is obtained with $\xi = 0$ and $\mathcal{F}_c = \mathcal{F}_m$.

C. Unified cylinder projection

As shown in [17] and illustrated in Fig. 4, one can compute the interpretation planes (${}^m\mathbf{n}_1$ and ${}^m\mathbf{n}_2$) of the two edges of a cylinder from the binormalized Plücker coordinates of the cylinder axis (${}^m\mathbf{u}$, ${}^m\mathbf{n}$):

$$\begin{cases} {}^m\mathbf{n}_1 = \cos\theta {}^m\mathbf{n} - \sin\theta {}^m\mathbf{u} \times {}^m\mathbf{n} \\ {}^m\mathbf{n}_2 = -\cos\theta {}^m\mathbf{n} - \sin\theta {}^m\mathbf{u} \times {}^m\mathbf{n} \end{cases} \quad (3)$$

where $\cos\theta = {}^m z / {}^m n$, $\sin\theta = R / {}^m n$, ${}^m z = \sqrt{{}^m n^2 - R^2}$ and R is the cylinder radius.

Therefore, using the previous subsection, the projection of a cylinder in a central catadioptric image is made of two conic curves ${}^p\Omega_1$ and ${}^p\Omega_2$ respectively associated to the two interpretation planes ${}^m\mathbf{n}_1$ and ${}^m\mathbf{n}_2$.

III. KINEMATIC MODELLING

A. Vision-based kinematics

The mechanism to calibrate is a Gough-Stewart platform (Fig. 1), *i.e.* a closed-loop mechanism in which the end-effector is connected to the base by 6 legs of varying length q_j , $j \in 1..6$. Each leg (Fig. 5) is attached to the base by a spherical joint located in point \mathbf{A}_j and to the moving platform (end-effector) by a universal or spherical joint located in point \mathbf{B}_j . The inverse kinematic model of such an hexapod is given by [2]

$$\forall j \in 1..6, \quad q_j^2 = \overline{\mathbf{A}_j \mathbf{B}_j}^T \overline{\mathbf{A}_j \mathbf{B}_j} \quad (4)$$

expressing that q_j is the length of vector $\overline{\mathbf{A}_j \mathbf{B}_j}$. This model can be expressed in any Euclidean reference frame. Hence, it can be expressed in the mirror frame \mathcal{R}_m .

Denoting ${}^m\mathbf{u}_j$ the unit vector pointing from ${}^m\mathbf{A}_j$ to ${}^m\mathbf{B}_j$, one can replace (4) by

$$q_j {}^m\mathbf{u}_j = {}^m\mathbf{R}_e {}^e\mathbf{B}_j + {}^m\mathbf{t}_e - {}^m\mathbf{A}_j \quad (5)$$

Thus, according to [17], one obtains the differential inverse kinematic model of the hexapod, relating the Cartesian velocity ${}^m\tau_m$ of the mirror frame, considered as attached to the base frame and seen from the end-effector, to the joint velocities by:

$$\dot{\mathbf{q}} = {}^m\mathbf{D}_m^{inv} {}^m\tau_m \quad (6)$$

with

$${}^m\mathbf{D}_m^{inv} = \begin{pmatrix} {}^m\mathbf{u}_1^T & ({}^m\mathbf{A}_1 \times {}^m\mathbf{u}_1)^T \\ \vdots & \vdots \\ {}^m\mathbf{u}_6^T & ({}^m\mathbf{A}_6 \times {}^m\mathbf{u}_6)^T \end{pmatrix} \quad (7)$$

where the ${}^m\mathbf{A}_j$ are constant calibration parameters.

B. Parameters necessary to vision-based control

The position and orientation in space of the legs are entirely defined by the joint locations on the base (\mathbf{A}_j , $j = 1..6$) and the leg directions (\mathbf{u}_j , $j = 1..6$) and entirely define the end-effector pose with respect to the base frame. Hence, vision-based control can be derived which depends only on the latter locations and directions and their time derivatives, for instance by servoing the leg directions as in [17].

In the case of a fixed camera with respect to the base, this only requires to use the interaction matrix associated to a leg direction given in [17] and expressed here in the mirror frame:

$${}^m\dot{\mathbf{u}}_j = \mathbf{L}_{\mathbf{u}_j}^T {}^m\tau_m \quad (8)$$

with

$$\mathbf{L}_{\mathbf{u}_j}^T = \frac{(\mathbf{I}_3 - {}^m\mathbf{u}_j {}^m\mathbf{u}_j^T)}{q_j} \left(\mathbf{I}_3 \quad -[{}^m\mathbf{A}_j + q_j {}^m\mathbf{u}_j] \times \right) \quad (9)$$

Consequently, the control law depends only on the attachment points of the legs onto the base expressed in the mirror frame (${}^m\mathbf{A}_j$), the joint offsets (q_{j0}) and the edges extracted from the image. However, considering the order of magnitude of ${}^m\mathbf{A}_j$ and q_j , one can neglect small errors on the joint offsets, that can hence be manually measured. Therefore, the kinematic parameters to be calibrated are reduced to ${}^m\mathbf{A}_j$.

IV. CALIBRATION METHOD

Now, the problem is to find ${}^m\mathbf{A}_j$ from the omnidirectional images. To do so, one first needs to extract adequate information from images and then to use this information in the calibration equations.

A. Visual information extraction

Once a conic curve ${}^p\Omega$ is fitted onto the image points, one can easily estimate the associated interpretation plane ${}^m\hat{\mathbf{n}}$. Indeed, inverting (2) gives an estimate ${}^m\hat{\Omega}$ of ${}^m\Omega$.

With the notation ${}^m\Omega = ({}^m\omega_1, {}^m\omega_2, {}^m\omega_3)$, one trivially gets from (1),

$${}^m\hat{\mathbf{n}} = \frac{{}^m\omega_3}{\|{}^m\omega_3\|} \quad (10)$$

Applying this procedure to the conic curves ${}^p\Omega_j^1$ and ${}^p\Omega_j^2$, projection of the j th leg in the omnidirectional image, we obtain the associated estimated interpretation planes ${}^m\hat{\mathbf{n}}_j^1$ and ${}^m\hat{\mathbf{n}}_j^2$.

Then, having the estimates (${}^m\hat{\mathbf{n}}_1$ and ${}^m\hat{\mathbf{n}}_2$) of the two cylinder edges, one can then reconstruct the cylinder direction by:

$${}^m\hat{\mathbf{u}} = \frac{{}^m\hat{\mathbf{n}}_1 \times {}^m\hat{\mathbf{n}}_2}{\|{}^m\hat{\mathbf{n}}_1 \times {}^m\hat{\mathbf{n}}_2\|} \quad (11)$$

to feed the control law.

B. Calibration process

The control model depends on the attached points of the legs onto the base expressed in the camera frame ${}^m\mathbf{A}_j$ as kinematics parameters. Assuming that this attachment point is lying on the revolution axis of the leg with radius R (Fig. 4), one can easily show, for any leg j and robot configuration k ,

$$({}^m\hat{\mathbf{n}}_1^{j,k})^T {}^m\mathbf{A}_j = -R \quad (12)$$

$$({}^m\hat{\mathbf{n}}_2^{j,k})^T {}^m\mathbf{A}_j = -R \quad (13)$$

Consequently, for N robot configurations, we can build the following linear system for each leg j from the image information:

$$\begin{pmatrix} ({}^m\hat{\mathbf{n}}_1^{j,1})^T \\ ({}^m\hat{\mathbf{n}}_2^{j,1})^T \\ \vdots \\ ({}^m\hat{\mathbf{n}}_1^{j,N})^T \\ ({}^m\hat{\mathbf{n}}_2^{j,N})^T \end{pmatrix} {}^m\mathbf{A}_j = \begin{pmatrix} -R \\ -R \\ \vdots \\ -R \\ -R \end{pmatrix} \quad (14)$$

This system has a unique least-square solution if there are at least two configurations with different leg directions. To improve its numerical efficiency, one uses robot configurations with the larger angles between each leg direction.

The calibration procedure is hence reduced to a strict minimum. However, to deal with noisy estimates due to image noise, it is preferable to solve for (14) using robust linear regression (such as iterative weighted least squares, M-estimators, or median least squares) rather than using bare linear algebra.

Let us finally remark that this method estimates each attachment point independently from the others and does not make use of any knowledge on the base geometry. There are two ways to handle the latter. The first one consists in solving simultaneously (14) together with constraints linking the attachment points, but then one loses the linear property. Alternately, we prefer to estimate independently the attachment points and then to fit the best rigid transformation between the robot base and the mirror from the estimated attachment points in the mirror frame and their CAD values:

$${}^m\hat{\mathbf{T}}_b = \arg \min_{{}^m\mathbf{T}_b} \frac{1}{2} \sum_{j=1}^6 \|\widehat{{}^m\mathbf{A}}_j - {}^m\mathbf{T}_b {}^b\mathbf{A}_j^{CAD}\|^2 \quad (15)$$

Notice that a scale factor error on R yields the same scale factor error on the solution of (14). Such an error can be compensated for *a posteriori* using a global scale factor such as the mean ratio r between the inter-point distances computed from the joint location estimates and their CAD values:

$$r = \text{mean}_{j \neq k} \frac{\|\widehat{{}^m\mathbf{A}}_j - \widehat{{}^m\mathbf{A}}_k\|}{\|{}^b\mathbf{A}_j^{CAD} - {}^b\mathbf{A}_k^{CAD}\|} \quad (16)$$

V. EXPERIMENTAL RESULTS

A. Joint center estimation

In a first experiment, we applied the calibration method to a catadioptric camera placed between the legs, with the 640x480 pixel retina parallel to the base plane (Fig. 1, using 60 robot configurations located on the workspace limits. A qualitative evaluation of the results is to reproject the estimated joint locations ${}^m\hat{\mathbf{A}}_j$ onto the image plane. Fig. 6 superimposes them (yellow circles) onto the image. This shows that the estimation is very correct in the horizontal plane, even though no calibration pattern was used for calibration but only the appearance of the robot itself.

A slightly more quantitative way to evaluate the accuracy is to plot (see Fig. 7) the estimated joint locations (red) and their CAD value (blue) in a common reference frame, *i.e.* after solving (15). This shows that the estimation errors are mainly along the vertical which is coherent with the image reprojection in Fig. 6, but are nevertheless smaller than 1 cm.

Finally, the square root of the median square error between the CAD geometry and the reconstructed one is 1.2 cm. This result is highly satisfying with regard to the omnidirectional camera poor resolution and the fact that no specific visual pattern was used.

In a second experiment, the robot is observed by a 1024x780 pixel perspective camera placed horizontally in front of it (Fig. 2). Since this case is a particular case of the unifying model with $\mathcal{F}_m = \mathcal{F}_c$, the left superscript c is used rather than m to distinguish from an actual omnidirectional camera.

Since the perspective case is physically simpler, we could also estimate at hand, yet carefully, the estimation of the transformation matrix ${}^c\mathbf{T}_b$ between the base frame and the camera frame. This allows to build manually estimated joint locations from the CAD model ${}^c\mathbf{A}_j^{hand}$.



Fig. 6. Reprojection of the estimated joint locations ${}^m\widehat{\mathbf{A}}_j$ when using an omnidirectional camera.

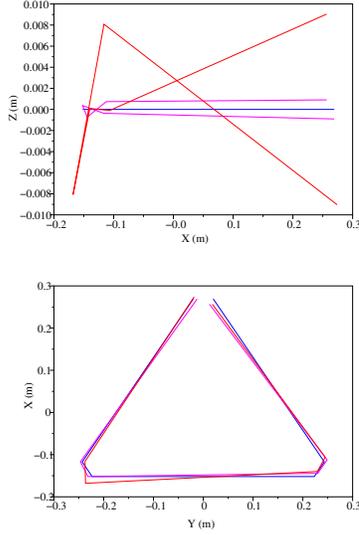


Fig. 7. Estimated geometry (in meters) in the omnidirectional case (red) and perspective case (magenta) compared to the CAD geometry (blue): lateral view (top) and top view (bottom).

Note that this manual estimation is troublesome since the perspective camera frame can not be accurately materialized. It is almost unrealistic in the omnidirectional case, since one would have to find the mirror frame which is attached to an immaterial unit sphere.

Fig. 8 shows the reprojection of the joint locations ${}^c\widehat{\mathbf{A}}_j$ in the image estimated through the proposed calibration method (circles) as well as the manually estimated ones ${}^c\mathbf{A}_j^{hand}$ (crosses). Hence, the proposed method yields an accuracy increase, compared to manual estimation.

The reconstructed geometry is also plotted (magenta) in Fig. 7. It has a better estimation in the vertical direction than in the omnidirectional case, since the perspective image plane is vertical. The deformation in the horizontal plane is slightly smaller than in the omnidirectional case, probably because the resolution of the perspective camera is twice the one of the omnidirectional camera.

Quantitatively, the square root of the median square error between the CAD geometry and the reconstructed one is now



Fig. 8. Reprojection of the estimated joint locations ${}^c\widehat{\mathbf{A}}_j$ (circles) when using a perspective camera and of the manually estimated ones ${}^c\mathbf{A}_j^{hand}$ (crosses).

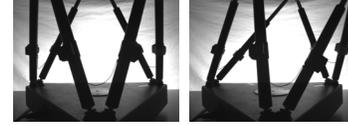


Fig. 9. Initial (left) and desired (right) robot configurations.

0.8 cm, which is also satisfying.

Recall that the attachment points of the robot legs onto the base are only appearing in the interaction matrix in edge-based control, but not in the servoed error. Therefore, the accuracy obtained above is enough to ensure the control stability.

B. The effects of the kinematics parameters on the control law

We achieved the visual servoing of an hexapod using a perspective camera (Fig. 9). Therefore, we can evaluate the calibration results in view of their use at control time, which is actually the final goal. However, we restrict this validation to the perspective case, since the control using the omnidirectional camera is not achieved yet.

In the reported experience, the perspective camera is fixed with respect to the base in the same position of calibration process (Fig. 2). Fig. 10 presents the norm of the regulated error when using manually estimated joint locations ${}^c\mathbf{A}_j^{hand}$ (error curve \mathbf{E}_3) and when using the joint locations ${}^c\widehat{\mathbf{A}}_j$ estimated with the proposed robust linear method (error curve \mathbf{E}_2). Additionally, a third run of the control was performed using the CAD geometry and the estimated camera-to-base transformation (15) (error curve \mathbf{E}_1), *i.e.* ${}^c\mathbf{A}_j \approx {}^c\widehat{\mathbf{T}}_b^c \mathbf{A}_j^{CAD}$.

In both cases, control converges to 0. This is normal since the calibration parameters (and the associated errors) do not appear in the error signal.

In both cases also, exponential convergence is obtained in the strict theoretical meaning: the error curves are bounded by an exponential decaying curve. However, when the attachment points are estimated at hand, an overshoot appears which is drastically reduced when they are calibrated.

Moreover, one can not clearly distinguish between the calibrated results, with or without taking into account the CAD model. Therefore, the robust linear fitting is enough for control, and no additional computational cost is needed.

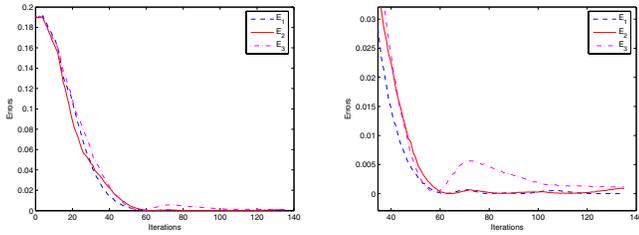


Fig. 10. Evolution of the edge-based control: total evolution (left) and zoom on the convergence tail (right).

VI. CONCLUSION

In this paper, a method for calibrating a Gough-Stewart parallel robot in view of its vision-based control was proposed. It estimates the attachment points of the robot legs onto the base, that are the only kinematic parameters the control really depends on. To do so, rather than placing a calibration pattern onto the mechanism, we used the direct observation of the legs.

This method works in the case where an omnidirectional camera is used as well as in the case where a single perspective camera is used. However, using an omnidirectional camera overcomes the self-occlusion problem arising in the perspective case.

Experimental validation of the method was presented in both cases, yielding an approximate 1 cm median error. This accuracy was demonstrated sufficient for perspective vision-based control. We believe it will also be sufficient for omnidirectional vision-based control, which will be implemented in the near future.

ACKNOWLEDGMENT

The authors would like to acknowledge the use of ViSP [32] in the perspective case. This study was jointly funded by CPER Auvergne 2003-2005 program and by the EU-IP project NEXT: NMP2-CT-2005-011815.

REFERENCES

- [1] B. Dasgupta and T.S. Mruthyunjaya. Force redundancy in parallel manipulators: theoretical and practical issues. *Mech. Mach. Theory*, 33(6):727–742, 1998.
- [2] J.P. Merlet. *Parallel robots*. Kluwer Academic Publishers, 2000.
- [3] J. Wang and O. Masory. On the accuracy of a Stewart platform - Part I : The effect of manufacturing tolerances. In *Proc. ICRA93*, pages 114–120, 1993.
- [4] D. Daney. Self calibration of Gough platform using leg mobility constraints. In *Proceedings of the 10th world congress on the theory of machine and mechanisms*, pages 104–109, Oulu, Finland, 1999.
- [5] W. Khalil et S. Besnard. Self calibration of stewart-gough parallel robots without extra sensors. *IEEE Transactions on Robotics and Automation*, 15:1116–1121, 1999.
- [6] C.W. Wampler, J.M. Hollerbach, and T. Arai. An implicit loop method for kinematic calibration and its application to closed-chain mechanisms. *IEEE Transactions on Robotics and Automation*, 11(5):710–724, 1995.
- [7] L. Tencredi, M. Teillaud, and J.P. Merlet. Forward kinematics of a parallel manipulator with additional rotary sensors measuring the position of platform joints. *Computational Kinematics, J.P. Merlet and B. Ravani, Eds., Dordrecht*, pages 261–270, December 1995.
- [8] M. Husty. An algorithm for solving the direct kinematics of general Gough-Stewart platforms. *Mech. Mach. Theory*, 31(4):365–380, 1996.

- [9] P. Dietmaier. The stewart-gough platform of general geometry can have 40 real postures. *Advances in Robot Kinematics: Analysis and Control, J. Lenarcic and M.L. Husty, Editors, Kluwer*, pages 1–10, 1998.
- [10] G. Gogu. Fully-isotropic T3R1-type parallel manipulator. In J. Lenarčič and C. Galletti, editors, *On Advances in Robot Kinematics*, pages 265–272. Kluwer Academic Publishers, 2004.
- [11] L. Baron and J. Angeles. The on-line direct kinematics of parallel manipulators under joint-sensor redundancy. In *Advances in Robot Kinematics*, pages 126–137. Kluwer Academic Publishers, 1998.
- [12] B. Espiau, F. Chaumette, and P. Rives. A New Approach To Visual Servoing in Robotics. *IEEE Trans. on Robotics and Automation*, 8(3), June 1992.
- [13] M.L. Koreichi, S. Babaci, F. Chaumette, G. Fried, and J. Pontnau. Visual servo control of a parallel manipulator for assembly tasks. In *6th Int. Symposium on Intelligent Robotic Systems, SIRS'98*, pages 109–116, Edimburg, Scotland, July 1998.
- [14] H. Kino, C.C. Cheah, S. Yabe, S. Kawamura, and S. Arimoto. A motion control scheme in task oriented coordinates and its robustness for parallel wire driven systems. In *Int. Conf. Advanced Robotics (ICAR'99)*, pages 545–550, Tokyo, Japan, Oct. 25-27 1999.
- [15] P. Kallio, Q. Zhou, and H. N. Koivo. Three-dimensional position control of a parallel micromanipulator using visual servoing. In Bradley J. Nelson and Editors Jean-Marc Breguet, editors, *Microrobotics and Microassembly II, Proceedings of SPIE*, volume 4194, pages 103–111, Boston, USA, November 2000.
- [16] C. Samson, M. Le Borgne, and B. Espiau. *Robot Control : The Task Function Approach*. Clarendon Press, Oxford, 1991.
- [17] N. Andreff and P. Martinet. Visual servoing of a Gough-Stewart parallel robot allows for reduced and linear kinematic calibration. In *Int. Conf. on Informatics in Control, Automation and Robotics (ICINCO'05)*, pages 119–124, Barcelona, Spain, 2005.
- [18] D. Stewart. A platform with six degrees of freedom. In *Proc. IMechE (London)*, volume 180, pages 371–386, 1965.
- [19] V.E. Gough and S.G. Whitehall. Universal tyre test machine. In *Proc. FISITA 9th Int. Technical Congress*, pages 117–137, May 1962.
- [20] P. Blaer and P.K. Allen. Topological mobile robot localization using fast vision techniques. In *IEEE International Conference on Robotics and Automation*, pages 1031–1036, Washington, USA, May 2002.
- [21] N. Winter, J. Gaspar, G. Lacey, and J. Santos-Victor. Omnidirectional vision for robot navigation. In *Proc. IEEE Workshop on Omnidirectional Vision, OMNIVIS*, pages 21–28, South Carolina, USA, June 2000.
- [22] J. P. Barreto, F. Martin, and R. Horaud. Visual servoing/tracking using central catadioptric images. In *ISER2002 - 8th International Symposium on Experimental Robotics*, pages 863–869, Bombay, India, July 2002.
- [23] Y. Mezouar, H. Hadj Abdelkader, P. Martinet, and F. Chaumette. Central catadioptric visual servoing from 3d straight lines. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'04*, pages 343–348, Sendai, Japan, September 28 October 2 2004.
- [24] H. Hadj Abdelkader, Y. Mezouar, N. Andreff, and P. Martinet. Image-based control of mobile robot with central catadioptric cameras. In *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'05*, pages 3533–3538, Barcelona, Spain, April 2005.
- [25] R. Vidal, O. Shakernia, and S. Sastry. Formation control of nonholonomic mobile robots with omnidirectional visual servoing and motion segmentation. In *IEEE International Conference on Robotics and Automation*, pages 584–589, Taipei, Taiwan, September 2003.
- [26] R. Benosman and S. Kang. *Panoramic Vision*. Springer Verlag ISBN 0-387-95111-3, 2000.
- [27] S. Baker and S. K. Nayar. A theory of single-viewpoint catadioptric image formation. *International Journal of Computer Vision*, 35(2):1–22, November 1999.
- [28] C. Geyer and K. Daniilidis. A unifying theory for central panoramic systems and practical implications. In *European Conference on Computer Vision*, volume 29, pages 159–179, Dublin, Ireland, May 2000.
- [29] J. Barreto and H. Araujo. Geometric properties of central catadioptric line images. *7th European Conference on Computer vision, ECCV'02, Copenhagen, Denmark*, pages 237–251, May 2002.
- [30] J. Plücker. On a new geometry of space. *Philosophical Transactions of the Royal Society of London*, 155:725–791, 1865.
- [31] N. Andreff, B. Espiau, and R. Horaud. Visual servoing from lines. *Int. Journal of Robotics Research*, 21(8):679–700, 2002.
- [32] E. Marchand, F. Spindler, and F. Chaumette. ViSP: A generic software platform for visual servoing. *IEEE Robotics and Automation Magazine*, 12(4):40–52, December 2005.