Visual Servoing of a Gough-Stewart Parallel Robot without Proprioceptive Sensors

Nicolas Andreff, Philippe Martinet

Visual Servoing of a Gough-Stewart Parallel Robot without Proprioceptive Sensors

Nicolas Andreff and Philippe Martinet
LASMEA - CNRS - Université Blaise Pascal/IFMA, 63175 Aubière, France
Email: {andreff|martinet}@lasmea.univ-bpclermont.fr
http://wwwlasmea.univ-bpclermont.fr/Control

Abstract

In this paper, it is shown that computer vision, used as a redundant metrology mean, allows the control of a Gough-Stewart parallel robot without using any joint sensor. This result is highly relevant for the control of parallel mechanisms hard to instrument (for instance, hydraulically actuated ones) or with large displacements (for instance, large telescopes positioning systems). It is also very interesting since it turns computer vision, usually considered as an exteroceptive sensor, into a proprioceptive one, yet preserving its exteroceptive nature and non contact measurement.

1 Introduction

Parallel mechanism are such that there exist several kinematic chains (or legs) between their base and their end-effector. Therefore, they may exhibit a better repeatability [21] than serial mechanisms but not a better accuracy [23], because of the large number of links and passive joints. There can be two ways to compensate for the low accuracy. The first way is to perform a kinematic calibration of the mechanism and the second one is to use a control law which is robust to calibration errors.
There exists a large amount of work on the control of parallel mechanisms (see [1] for a long list of references). In the focus of attention, Cartesian control is naturally achieved through the use of the inverse Jacobian which transforms Cartesian velocities into joint velocities. It is noticeable that the inverse Jacobian of parallel mechanisms does not only depend on the joint configuration (as for serial mechanisms) but also on the end-effector pose.
Consequently, one needs to be able to estimate or measure the latter. As far as we know, all the effort has been put on the estimation of the end-effector pose through the forward kinematic model and the joint measurements. However, this yields much trouble, related to the fact that there is usually no analytic formulation of the forward kinematic model of a parallel mechanism. Hence, one numerically inverts the inverse kinematic model, which is analytically defined for most of the parallel mechanisms. However, it is known [20, 12] that this numerical inversion requires high order polynomial root determination, with several possible solutions (up to 24 real solutions for a Gough-Stewart platform). Much of the work is thus devoted to solving this problem accurately and in real-time (see for instance [25]), or to designing parallel mechanisms with analytical forward kinematic model [14, 10]. One of the promising paths lies in the use of the so-called metrological redundancy [3], which simplifies the kinematic models by introducing additional sensors into the mechanism and thus yields easier control [19].
Computer vision being an efficient way of estimating the end-effector pose [7, 16], it is a good alternative to use it for Cartesian control of parallel mechanisms. It can be done in three ways.

Vision as a sensor The first one consists in computing the end-effector poses by vision, then in translating them into joint configurations, through the inverse kinematic model, and finally servoing in the joint space. However, such schemes should be used carefully for parallel mechanisms, since joint control does not take into account the kinematic closures and may therefore yield high internal forces [6].

Visual servoing Second, vision can be additionally used to perform visual servoing [8, 5]. Indeed, instead of measuring the end-effector pose and convert it into joint values, one can directly use the image for control. Visual servoing techniques are very effective since they close the control loop on the vision sensor, which gives a direct view of the Cartesian space. This yields a high

robustness to robot calibration errors. Indeed, these errors only appear in a Jacobian matrix but not in the regulated error.
Essentially, these techniques generate a Cartesian desired velocity which is converted into joint velocities by the inverse Jacobian. Hence, one can translate such techniques to parallel mechanisms. It should even be rather easier than in the serial case, since the inverse Jacobian of a parallel mechanism is usually analytical. Visual servoing techniques have been applied to the control of parallel mechanisms with a reduced number of degrees of freedom in [15, 13].

A novel approach However, these previous two ways consist solely in a simple adaptation of now classical control schemes, which, although probably very efficient, are not very innovative. Therefore, a novel third way to use vision, which gathers the advantages of redundant metrology and of visual servoing and avoids most of their drawbacks was presented in [2].
Indeed, adding redundant sensors is not always technically feasible (think of a spherical joint) and always requires either that the sensors are foreseen at design stage or that the mechanism is physically modified to install them after its building. Anyhow, there are then additional calibration parameters in the kinematic model and one needs to estimate them in order to convert redundant joint readings into a unit vector expressed in the appropriate reference frame. Moreover, observing the end-effector of a parallel mechanism by vision may be incompatible with its application. For instance, it is not wise to imagine observing the end-effector of a machining tool. On the opposite, it should not be a problem to observe the legs of the mechanism, even in such extreme cases. Thereby one would turn vision from an exteroceptive sensor to a somewhat more proprioceptive sensor. This brings us back to the redundant metrology paradigm.
Extending the idea of vision as a proprioceptive sensor to its limit, we propose in this paper a visual servoing scheme which allows to control a Gough-Stewart parallel robot [11, 22] (Figure 1) without using any joint sensor information. This contribution seems especially relevant to the control of parallel robots that are hard to instrument (for instance, hydraulically actuated ones) or that have large displacements (for instance, large telescopes positioning systems). Moreover, it will be seen that such a technique is easy to implement, on the opposite to the visual servoing of non-instrumented serial robots [18] where tedious programming was necessary.
The remainder of the paper is the following. Section 2 recalls the vision-based kinematics of the Gough-Stewart robot. Then, section 3 recalls the differential geometry aspect of the leg observation and presents the joint-free control derived from it. Finally, simulation results and conclusion are given respectively in section 4 and section 5.

Figure 1: A Gough-Stewart platform observed by a camera with short focal length.

2 Vision-based kinematics

Consider the hexapod in Figure 1. It has 6 legs of varying length $q_i, i \in 1..6$, attached to the base by spherical joints located in points $\mathbf{A}_i$ and to the moving platform (end-effector) by spherical joints located in points $\mathbf{B}_i$. The inverse kinematic model of such an hexapod is given by

$$\forall i \in 1..6, \quad q_i^2 = \overrightarrow{\mathbf{A}_i\mathbf{B}_i}^T \overrightarrow{\mathbf{A}_i\mathbf{B}_i} \tag{1}$$

expressing that q_i is the length of vector $\overrightarrow{\mathbf{A}_i\mathbf{B}_i}$. This model can be expressed in any Euclidean reference frame. Hence, it can be expressed in the base frame $\mathcal{R}_b$, in the end-effector frame $\mathcal{R}_e$ or in the camera frame $\mathcal{R}_c$. In the remainder, the camera reference frame will be used and denoted by a left upper-script. According to [2] and assuming a calibrated camera, one can express the vision-based kinematics of the hexapod expressed in the camera frame:

$$q_i\, {}^c\underline{\mathbf{u}}_i = {}^c\mathbf{R}_e\, {}^e\mathbf{B}_i + {}^c\mathbf{t}_e - {}^c\mathbf{A}_i \tag{2}$$

$$\dot{\mathbf{q}} = {}^c\mathbf{J}_c^{inv}\, {}^c\tau_c \tag{3}$$

with

$${}^c\mathbf{J}_c^{inv} = -\begin{bmatrix} {}^c\underline{\mathbf{u}}_1^T & ({}^c\mathbf{A}_1 \times {}^c\underline{\mathbf{u}}_1)^T \\ \vdots & \vdots \\ {}^c\underline{\mathbf{u}}_6^T & ({}^c\mathbf{A}_6 \times {}^c\underline{\mathbf{u}}_6)^T \end{bmatrix} \tag{4}$$

where ${}^c\tau_c$ is the Cartesian velocity of the camera frame, considered as attached to the base frame and moving with respect to a fixed end-effector, expressed in itself and ${}^c\underline{\mathbf{u}}_i, i = 1..6$ are the unit vectors giving the pointing direction of each leg in the camera frame. Under the assumption that the legs are cylinders, those vectors can be easily detected as the intersection of the two cylinder edges in the image, while tracking such edges is rather easy [17].

3 Visual servoing of the platform

In this section, the control problem is addressed: from a given configuration of the hexapod legs observed by a camera attached to the base, how to reach a desired configuration ?
Visual servoing is based on the so-called interaction matrix $\mathbf{L}^T$ [4] which relates the instantaneous relative motion $T_c = {}^c\tau_c - {}^c\tau_s$ between the camera and the scene, to the time derivative of the vector s of all the visual primitives that are used through:

$$\dot{s} = \mathbf{L}_{(s)}^T T_c \tag{5}$$

where ${}^c\tau_c$ and ${}^c\tau_s$ are respectively the kinematic screw of the camera and the scene, both expressed in $\mathcal{R}_c$.
Then, one achieves exponential decay of an error $e(s, s_d)$ between the current primitive vector s and the desired one s_d using a proportional linearizing and decoupling control scheme of the form:

$$T_c = -\lambda \hat{\mathbf{L}}_{(s)}^{T+} e(s, s_d) \tag{6}$$

where T_c is used as a pseudo-control variable.
Here also, we will need to define a visual primitive, then form an error between its current value and its desired one, then relate in some way its time derivative to the actuation, and finally find a control relation between the error and the actuation.

3.1 Visual primitive and error

As foreseen above, the unit vectors ${}^c\underline{\mathbf{u}}_i, i = 1..6$ will be used as visual primitives. The visual primitives being unit vectors, it is theoretically more elegant to use the geodesic error rather than the standard vector difference. Consequently, the error grounding the proposed control law will be

$$\mathbf{e}_i = {}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{u}}_{di} \tag{7}$$

3.2 Interaction matrix

With the same convention as in (3), one has [2] the following interaction matrix for each leg

$$\begin{aligned} {}^c\dot{\underline{\mathbf{u}}}_i &= \frac{1}{q_i}\mathbf{M}_i^T\, {}^c\tau_c & (8)\\ \mathbf{M}_i^T &= -\left(\mathbf{I}_3 - {}^c\underline{\mathbf{u}}_i {}^c\underline{\mathbf{u}}_i^T\right)\begin{bmatrix}\mathbf{I}_3 & -[{}^c\mathbf{A}_i + q_i {}^c\underline{\mathbf{u}}_i]_\times\end{bmatrix} & (9)\end{aligned}$$

where $\mathbf{M}_i^T$ is 3×6 and is obviously of rank 2.
An interaction matrix $\mathbf{M}^T$ can then be obtained by stacking $\mathbf{M}_i^T, i = 1..6$. However, it is, in our opinion, an open question whether $\mathbf{M}$ shall or shall not be considered as an interaction matrix. Indeed, in visual servoing the various visual primitives are the image projections of objects in space that are rigidly linked to each other, while, here, each of the legs is in relative motion with respect to the other ones.
Nevertheless, effective control can be derived as shown in the following section.

3.3 Control law

Let us choose a control such that $E = (\mathbf{e}_1^T, ..., \mathbf{e}_6^T)^T$ decreases exponentially, i.e. such that

$$\dot{E} = -\lambda E \tag{10}$$

Then, introducing $\mathbf{N}_i^T = -[{}^c\underline{\mathbf{u}}_{di}]_\times \mathbf{M}_i^T$, $\mathbf{N}^T = (\mathbf{N}_1, ..., \mathbf{N}_6)^T$ and

$$\mathbf{\Delta}(\mathbf{q}) = \begin{bmatrix} \frac{1}{q_1}\mathbf{I}_3 & & \\ & \ddots & \\ & & \frac{1}{q_6}\mathbf{I}_3 \end{bmatrix}, \mathbf{\Delta}^{-1}(\mathbf{q}) = \begin{bmatrix} q_1\mathbf{I}_3 & & \\ & \ddots & \\ & & q_6\mathbf{I}_3 \end{bmatrix} \tag{11}$$

the combination of (7), (8) and (10) gives

$$\mathbf{\Delta}(\mathbf{q})\mathbf{N}^T\, {}^c\tau_c = -\lambda E \tag{12}$$

A control law using the joint sensor can be derived, giving the Cartesian control velocity:

$${}^c\tau_c = -\lambda \widehat{\mathbf{N}^T}^+ \mathbf{\Delta}^{-1}(\mathbf{q}) E \tag{13}$$

which can be transformed into the *admissible* control joint velocities using (3)

$$\dot{\mathbf{q}} = -\lambda \widehat{{}^c\mathbf{J}_c^{inv}}\, \widehat{\mathbf{N}^T}^+ \mathbf{\Delta}^{-1}(\mathbf{q}) E \tag{14}$$

where the *hat* means that only an estimate can be used.
Usually, in visual servoing schemes, people either choose either to estimate the interaction matrix at

each iteration using the latest measurements or to estimate it only once at the desired configuration. Following this idea, we chose to feed the interaction matrix with the measured ${}^c\underline{\mathbf{u}}_i$'s at each iteration. However, since the joint values appear marginally in (8) and (9), we can introduce a control scheme which *does not make any use of the joint values* by using the median joint values (*i.e.* we replace q_i with $q_{i_{med}} = (q_{i_{max}} + q_{i_{min}})/2$:

$$\dot{\mathbf{q}} = -\lambda \widehat{{}^c\mathbf{J}_c^{inv}} \widehat{\mathbf{N}^T}^{+}_{|\mathbf{q}=\mathbf{q}_{med}} \mathbf{\Delta}^{-1}(\mathbf{q}_{med}) E \qquad (15)$$

As early stated by Weiss *et al.* [24], the desired joint velocities can now be fed to an inner control loop closed around the joint sensors or directly sent in open loop to the actuators. In the latter case, the frame rate has to be high enough to cope with the actuators bandwidth. In practice, using standard off-the-shelf cameras, one can reach 100 images/second (i.e. 10 ms loop time), and with little extra effort, a 1 ms loop time can be attainable, see [9] for a 1kHz visual servoing of a serial robot using predictive control.

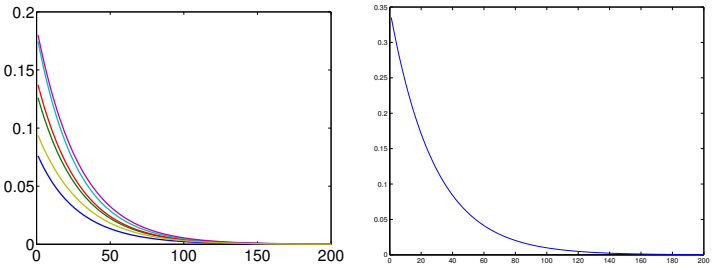
4 Results

A commercial DeltaLab hexapod was simulated, such that ${}^b\mathbf{A}_{2k} = R_b \begin{pmatrix} \cos(k\frac{\pi}{3}+\alpha) \\ \sin(k\frac{\pi}{3}+\alpha) \\ 0 \end{pmatrix}$, ${}^b\mathbf{A}_{2k+1} = R_b \begin{pmatrix} \cos(k\frac{\pi}{3}-\alpha) \\ \sin(k\frac{\pi}{3}-\alpha) \\ 0 \end{pmatrix}$, ${}^e\mathbf{B}_{2k} = R_e \begin{pmatrix} \cos(k\frac{\pi}{3}+\beta) \\ \sin(k\frac{\pi}{3}+\beta) \\ 0 \end{pmatrix}$, ${}^e\mathbf{B}_{2k+1} = R_e \begin{pmatrix} \cos(k\frac{\pi}{3}-\beta) \\ \sin(k\frac{\pi}{3}-\beta) \\ 0 \end{pmatrix}$, $k \in \{0,1,2\}$ with $R_b = 270mm$, $\alpha = 4.25°$, $R_e = 195mm$, $\beta = 5.885°$ and the legs range are $[345mm, 485mm]$.

4.1 Control validation

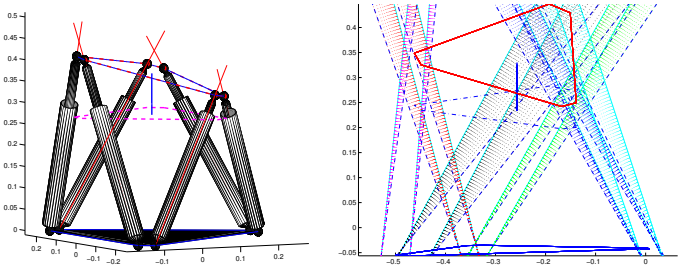
In all the simulations presented here and below, the initial configuration of the platform is the reference configuration where all the legs have minimal length. The goal configuration is obtained from this reference configuration by a translation by $10cm$ along the z axis of the platform (upward vertical) and a rotation of 15 around the x axis, thus reaching the workspace limit.

In a first simulation, the joint sensors are used for control according to (14). Figure 2 shows that the errors on each leg converges exponentially to 0 (in the strict theoretical meaning). This does not prevent some coupling to appear in the transient phase, which also appears in the joint velocities evolution (Figure 4(a)).



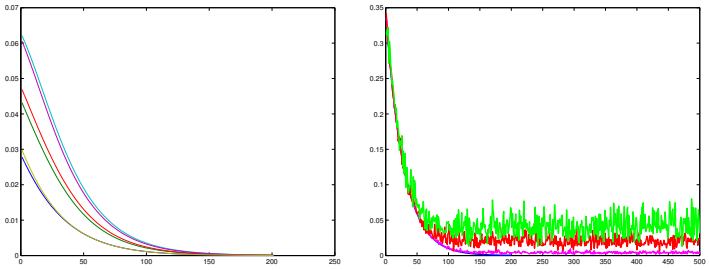
(a) (b)

Figure 2: Control using joint sensors: **(a)** Errors on each leg $\mathbf{e}_i^T\mathbf{e}_i$. **(b)** Sum of squared errors E^TE.



(a) (b)

Figure 3: Control using joint sensors: **(a)** Trajectory in space with initial (dashed) and desired (solid) position of the platform. **(b)** Image trajectories.

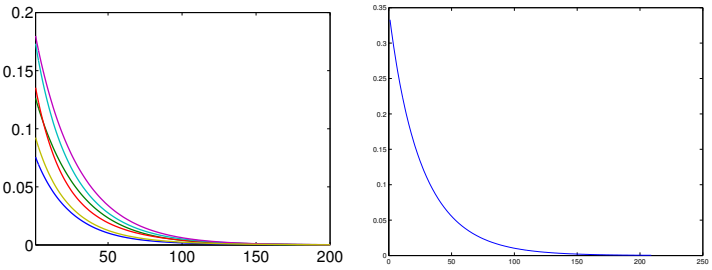


(a) (b)

Figure 4: Control using joint sensors: **(a)** Joint velocities. **(b)** Robustness to noise : sum of squares of the errors E^TE with a noise amplitude of 0 deg, 0.01 deg, 0.05 deg and 0.1 deg.

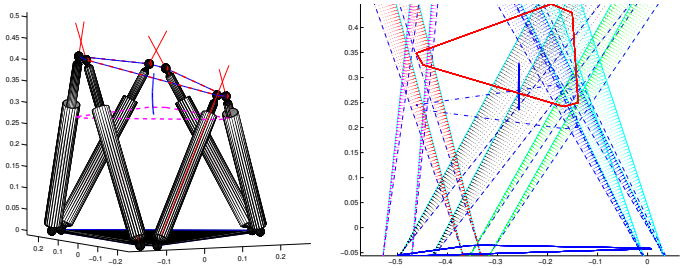
Figure 3(a) shows nevertheless that the desired end-effector pose is reached, as well as the desired image (Figure 3(b)). Noise was also added to the leg detection, as a random rotation of the interpretation planes defined by the optical center and the cylinder edges. Figure 4(b) shows that the control is rather robust to measurement errors.

In a second simulation, the joint sensors are not used any more for control according to (15). Figure 5 to 7 show a similar behaviour as in the case where the joint sensors are used. The only slightly visible difference is that each component of the error in Figure 5(a) has a different convergence rate (*i.e.* the exponentials cross each other) while they had the same in Figure 2. This phenomenon is hence found again on the joint velocity curves (Figure 7(a)).



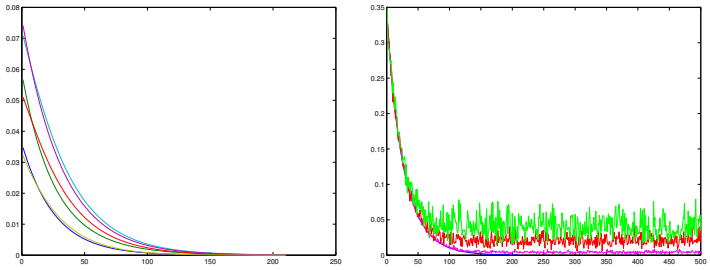
(a) (b)

Figure 5: Control **without** using joint sensors: **(a)** Errors on each leg $\mathbf{e}_i^T\mathbf{e}_i$. **(b)** Sum of squared errors E^TE.



(a) (b)

Figure 6: Control **without** using joint sensors: **(a)** Trajectory in space with initial (dashed) and desired (solid) position of the platform. **(b)** Image trajectories.



(a) (b)

Figure 7: Control **without** using joint sensors: **(a)** Joint velocities. **(b)** Robustness to noise : sum of squares of the errors E^TE with a noise amplitude of 0 deg, 0.01 deg, 0.05 deg and 0.1 deg.

5 Conclusion

A fundamentally novel approach was proposed for controlling a parallel mechanism using vision as a proprioceptive sensor, replacing the standard joint sensors. Thus, such a control law can be used for non-instrumented parallel robots. It thus comes in complement to previous work on the vision-based control of non-instrumented serial robots.

In a close future, this contribution shall be experimented. A careful attention will need be put on the image tracking of the legs. Since self-occlusions of the mechanism may appear, it will be necessary to deal with them, either by defining a vision system able to track the legs in any case (stereo-vision for instance) or by studying the robustness of the proposed control to occlusions.

Moreover, removing the proprioceptive sensors disables any inner control loop devoted to servo the actual joint velocities onto the desired ones. This imposes that the visual feedback should have a bandwidth compatible with the robot dynamics, which seems attainable with standard off-the-shelf cameras.

Further away, we will try to extend this control law to other kinds of parallel robots.

Acknowledgment

This study was jointly funded by CPER Auvergne 2003-2005 program and by the CNRS-ROBEA program through the MP2 project.

References

[1] http://www-sop.inria.fr/coprin/equipe/merlet.

[2] N. Andreff, A. Marchadier, and P. Martinet. Vision-based control of a Gough-Stewart parallel mechanism using legs observation. In *Proc. Int. Conf. Robotics and Automation (ICRA'05)*, Barcelona, Spain, May 2005. Accepted.

[3] 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, 1998.

[4] F. Chaumette. *La commande des robots manipulateurs, Traité IC2*, chapter Asservissement visuel, pages 105–150. Hermès, 2002.

[5] H.I. Christensen and P. Corke, editors. *Int. Journal of Robotics Research – Special Issue on Visual Servoing*, volume 22, October 2003.

[6] B. Dasgupta and T.S. Mruthyunjaya. Force redundancy in parallel manipulators: theoretical and practical issues. *Mech. Mach. Theory*, 33(6):727–742, 1998.

[7] D. DeMenthon and L. Davis. Model-based object pose in 25 lines of code. *Lecture Notes in Computer Science*, pages 335–343, 1992.

[8] 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.

[9] J. Gangloff and M. de Mathelin. High speed visual servoing of a 6 DOF manipulator using multivariable predictive control. *Advanced Robotics*, 17(10):993–1021, December 2003. Special issue: advanced 3D vision and its aplication to robotics.

[10] G. Gogu. Fully-isotropic T3R1-type parallel manipulator. In J. Lenarčič and C. Galletti, editors, *On Advances in Robot Kinematics*, volume Kluwer Academic Publishers, pages 265–272. 2004.

[11] V.E. Gough and S.G. Whitehall. Universal tyre test machine. In *Proc. FISITA 9th Int. Technical Congress*, pages 117–137, May 1962.

[12] M. Husty. An algorithm for solving the direct kinematics of general Gough-Stewart platforms. *Mech. Mach. Theory*, 31(4):365–380, 1996.

[13] P. Kallio, Q. Zhou, and H. N. Koivo. Three-dimensional position control of a parallel micromanipulator using visual servoing. In Bradley J. Nelson and Jean-Marc Breguet, editors, *Microrobotics and Microassembly II, Proceedings of SPIE*, volume 4194, pages 103–111, November 2000.

[14] H.S. Kim and L.-W. Tsai. Evaluation of a Cartesian parallel manipulator. In J. Lenarčič and F. Thomas, editors, *Advances in Robot Kinematics: Theory and Applications*. Kluwer Academic Publishers, June 2002.

[15] 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.

[16] JM. Lavest, M. Viala, and M. Dhome. Do we really need an accurate calibration pattern to achieve a reliable camera calibration. In *Proceedings of ECCV98*, pages 158–174, Freiburg, Germany, June 1998.

[17] E. Marchand and F. Chaumette. Feature tracking for visual servoing purposes. *Robotics and Autonomous Systems*, June 2005.

[18] E. Marchand, F. Chaumette, F. Spindler, and M. Perrier. Controlling an uninstrumented manipulator by visual servoing. *Int. J. of Robotics Research*, 21(7):635–648, July 2002.

[19] F. Marquet. *Contribution à l'étude de l'apport de la redondance en robotique parallèle*. PhD thesis, LIRMM - Univ. Montpellier II, October 2002.

[20] J-P. Merlet. An algorithm for the forward kinematics of general 6 d.o.f. parallel manipulators. Technical Report 1331, INRIA, November 1990.

[21] J.P. Merlet. *Parallel robots*. Kluwer Academic Publishers, 2000.

[22] D. Stewart. A platform with six degrees of freedom. In *Proc. IMechE (London)*, volume 180, pages 371–386, 1965.

[23] 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.

[24] L. E. Weiss, A. C. Sanderson, and C. P. Neuman. Dynamic sensor-based control of robots with visual feedback. *IEEE Journal of Robotics and Automation*, RA-3(5):404–417, October 1987.

[25] X. Zhao and S. Peng. Direct displacement analysis of parallel manipulators. *Journal of Robotics Systems*, 17(6):341–345, 2000.