2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Image-Based Visual Servoing of the I4R parallel robot without Proprioceptive Sensors

Tej Dallej [1], Nicolas Andreff [1,2] and Philippe Martinet [1,3]

Abstract— **This paper proposes a method to control an I4R parallel robot by the observation of its legs with a calibrated camera. We show that the control law depends only on the edges of its forearms extracted from the image and that no proprioceptive sensors are used. Indeed, the variables needed for control (namely, the directions of the arms and forearms as well as the positions of the wrists) can be reconstructed from the forearms edges. Experimental validation of the reconstruction is given and simulation of the control with realistic noises is performed showing the validity of the approach.**

I. INTRODUCTION

Visual servoing techniques [1], [2], [3] are a good alternative for the control of parallel mechanisms, since they close the control loop over the vision sensor. The fundamental assumption is that external sensing of the end-effector pose in the feedback signal replaces advantageously the forward kinematic model. Indeed, the perception models are simpler than the kinematic models and contain less unmodelled physical phenomena. Visual servoing was however seldom applied to parallel robotics [4], [5], [6], [7]. Those applications rely on 3D visual servoing [8] where the end-effector pose is indirectly measured and used for regulation. It is also shown in [7] that such a control can be made without any joint value measurement.

The main limitation of this approach is that it requires the estimation of the end-effector to tool transformation, the world to base transformation and the whole kinematic parameter set. Moreover, observing the end-effector of a parallel mechanism may be incompatible with its application. For instance, it is not wise to imagine observing the end-effector of a machine tool.

To overcome these problems, a new way to use vision which gathers the advantages of redundant metrology [9], [10] and of visual servoing is presented in [11]. This method proposes to control by vision the well-known six DOF Gough-Stewart platform [12], [13]. It has a reduced set of kinematic parameters and does not require any visual target. In that method, the leg orientations are chosen as visual primitives. The control law was based on their reconstruction from the image which might not be very accurate for intrinsic reasons.

This work was supported by European Community through the IP projet NEXT number 0011815.

[1] LASMEA, 24 avenue des Landais 63177 Aubière Cedex, France
firstname.lastname@lasmea.univ-bpclermont.fr
http://wwwlasmea.univ-bpclermont.fr/Control

[2] LAMI - IFMA, 63175 Aubière Cedex, France

[3] ISRC - Intelligent Systems Research Center, Sungkyunkwan University, Suwon, South Korea

Fig. 1. The I4R robot (from [15]).

To make control robust with respect to modelling errors and under the assumption of cylindrical legs, it is proposed in [14] to servo the leg edges rather than the leg orientations which improved the practical robustness by servoing the legs in the image. However, in the Gough-Stewart robot case, 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.

In this paper we extend the results of [14] to the I4R robot (Fig. 1 and [15]) which is part of the family derived from the H4 [16], [17], [18], which has a simpler architecture than the Gough-Stewart robot. In particular, using a camera fixed to the base, we can easily observe all cylindrical legs of the I4R robot. Moreover, this extension is made in order to keep the property of [7] that joint sensors are not needed.

As in the case of the Gough-Stewart platform, the I4R has cylindrical legs. Thus, those cylinders can be observed to extract, directly from the image, the leg edges and the latter can be used as visual primitives in the control. The expected advantages of this method is that it should not only reduce the kinematic parameter set but also simplify the kinematic models.

Therefore, the contribution of this paper is to present an image-based control of the I4R parallel robot by observing its leg edges without joint sensors using a camera fixed with respect to the base.

Section II recalls the description of I4R modelling. Section III recalls the vision-based framework for expressing the robot kinematics and control. In section IV we introduce the measurement process of parameters necessary to vision-based control. Then, the visual servoing method is presented in Section V and validated in Section VI. Conclusions are to be found in Section VII.

II. MECHANISM MODELING

The I4R parallel robot is based on four identical kinematic chains (Fig. 2). Each revolute motor, located in $\mathbf{P}_i$, moves

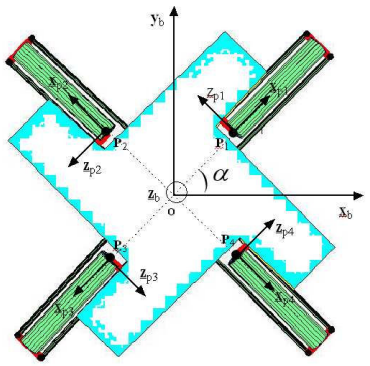

Fig. 2. I4R parameters.

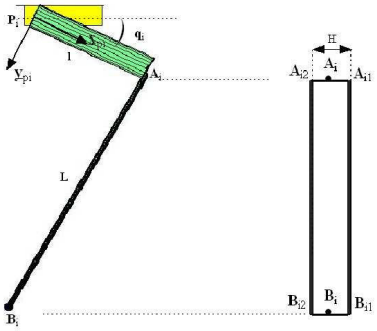

Fig. 3. Leg parameters.

an arm i (Fig. 3) attached in points $\mathbf{P}_i$ and $\mathbf{A}_i$. Each arm is connected to a forearm made of two legs equipped with ball joints $((\mathbf{A}_{i1}, \mathbf{A}_{i2})$ and $(\mathbf{B}_{i1}, \mathbf{B}_{i2}))$ forming a parallelogram. These forearms are connected, at each end, to the articulated travelling plate. This one supports the end-effector $\mathbf{E}$ that can be translated in three directions and rotated around a fixed axis ${}^c\underline{\mathbf{z}}_e$.

The end-effector of the I4R prototype is rotated by the relative displacement of the two plate parts (Fig. 4), using two rack-and-pinion systems. The relative translation $\mathbf{T}$ is transformed into a proportional end-effector rotation $\theta = \mathbf{T}/\mathcal{K}$.

Additional notation used throughout the paper are displayed in Table I.

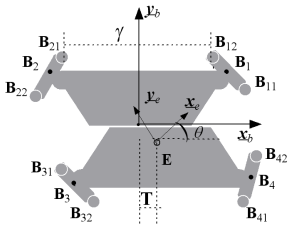

Fig. 4. I4R travelling plate parameters.

- $i = 1..4$ denotes the legs and $j = 1..2$ denotes the edges.
- Boldface lower-case characters denote vectors. Unit vectors are underlined.
- Capital boldface characters denote matrices.
- $\mathcal{F}_b = (\mathbf{O}, \underline{\mathbf{x}}_b, \underline{\mathbf{y}}_b, \underline{\mathbf{z}}_b)$, $\mathcal{F}_e = (\mathbf{E}, \underline{\mathbf{x}}_e, \underline{\mathbf{y}}_e, \underline{\mathbf{z}}_e)$, $\mathcal{F}_c = (\mathbf{O}_c, \underline{\mathbf{x}}_c, \underline{\mathbf{y}}_c, \underline{\mathbf{z}}_c)$ and $\mathcal{F}_{pi} = (\mathbf{P}_i, \underline{\mathbf{x}}_{pi}, \underline{\mathbf{y}}_{pi}, \underline{\mathbf{z}}_{pi})$ denote respectively the base, end-effector, camera and i^{th} arm reference frames.
- ${}^i\mathbf{v}$ is vector $\mathbf{v}$ expressed in $\mathcal{F}_i$.
- $\mathbf{K}$ is the matrix of the camera intrinsic parameters.
- q_i is the position of motor i.
- We express all kinematic parameters in the camera frame $\mathcal{F}_c$.

TABLE I

NOTATION USED THROUGHOUT THE PAPER.

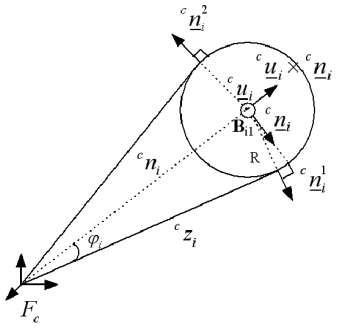

Fig. 5. Visual edges of a cylinder.

III. VISION-BASED KINEMATICS

A. Image Projection of a Leg

As shown in Fig. 3, each forearm i is made of two rods $[\mathbf{A}_{i1}\mathbf{B}_{i1}]$ and $[\mathbf{A}_{i2}\mathbf{B}_{i2}]$. We choose to observe only the first rod defined by $[\mathbf{A}_{i1}\mathbf{B}_{i1}]$ but the method can make use of both rods.

We assume that the attachment point $\mathbf{B}_{i1}$ is lying on the revolution axis of the leg with radius R. Consequently, a cylinder edge is defined by the following constraints expressed in the camera frame (Fig. 5):

$$\begin{cases} {}^c\underline{\mathbf{n}}_i^{j\,T}\,{}^c\underline{\mathbf{u}}_i = 0 \\ {}^c\underline{\mathbf{n}}_i^{j\,T}\,{}^c\underline{\mathbf{n}}_i^j = 1 \\ {}^c\mathbf{B}_{i1}{}^T\,{}^c\underline{\mathbf{n}}_i^j = -R \end{cases} \tag{1}$$

where ${}^c\underline{\mathbf{u}}_i$ is the direction of the revolution axis, ${}^c\underline{\mathbf{n}}_i^j$ is the unit vector normal to plane passing by the center of projection and tangent to the cylinder. The latter plane (known as interpretation plane) defines the image projection of the edge.

B. The Differential Inverse Kinematic Model

Referring to Fig. 3, we have:

$${}^c\overrightarrow{\mathbf{A}_{ij}\mathbf{B}_{ij}} = {}^c\overrightarrow{\mathbf{A}_{i1}\mathbf{B}_{i1}} = {}^c\overrightarrow{\mathbf{A}_{i2}\mathbf{B}_{i2}} = L\,{}^c\underline{\mathbf{u}}_i \tag{2}$$

If we observe only the edges of each rod $[\mathbf{A}_{i1}\mathbf{B}_{i1}]$, the expression of the kinematic chain closure yields the so-called

implicit kinematic model in vector form [19]:

$$L^c\underline{\mathbf{u}}_i = {}^c\overrightarrow{\mathbf{A}_{i1}\mathbf{B}_{i1}} = {}^c\mathbf{B}_{i1} - {}^c\mathbf{A}_{i1} \quad (3)$$

Notice that this expression is also available using the second leg $[\mathbf{A}_{i2}\mathbf{B}_{i2}]$ or a fictive leg $[\mathbf{A}_i\mathbf{B}_i]$ and that it is valid for any other reference frame.

Time differentiating (3) yields:

$$L^c\dot{\underline{\mathbf{u}}}_i = \frac{d}{dt}({}^c\mathbf{B}_{i1}) - \frac{d}{dt}({}^c\mathbf{A}_{i1}) \quad (4)$$

Referring to Fig. 3 again, one gets:

$${}^c\mathbf{A}_{i1} = {}^c\mathbf{P}_i + l^c\underline{\mathbf{x}}_{pi} + {}^c\overrightarrow{\mathbf{A}_i\mathbf{A}_{i1}} \quad (5)$$

The direction of each arm ${}^c\underline{\mathbf{x}}_{pi}$ is a unit vector rotating around the motor rotation axis ${}^c\underline{\mathbf{z}}_{pi}$ with velocity $\dot{q}_i$. Moreover, ${}^c\mathbf{P}_i$ and ${}^c\overrightarrow{\mathbf{A}_i\mathbf{A}_{i1}} = -\frac{1}{2}H^c\underline{\mathbf{z}}_{p_i}$ are constant parameters. Hence, one gets:

$$\frac{d}{dt}({}^c\mathbf{A}_{i1}) = l\frac{d}{dt}({}^c\underline{\mathbf{x}}_{pi}) = \dot{q}_i l^c\underline{\mathbf{y}}_{pi} \quad (6)$$

where ${}^c\underline{\mathbf{y}}_{p_i} = {}^c\underline{\mathbf{z}}_{p_i} \times {}^c\underline{\mathbf{x}}_{p_i}$ Now, using

$${}^c\mathbf{B}_{i1} = {}^c\mathbf{E} + {}^c\overrightarrow{\mathbf{E}\mathbf{B}_{i1}} \quad (7)$$

one can obtain:

$$\frac{d}{dt}({}^c\mathbf{B}_{i1}) = {}^cV_e + \frac{d}{dt}({}^c\overrightarrow{\mathbf{E}\mathbf{B}_{i1}}) \quad (8)$$

where ${}^cV_e$ is the translational velocity of the end-effector.

Taking into account the relative displacement of the two plate parts, one can easily obtain the time derivative of ${}^c\overrightarrow{\mathbf{E}\mathbf{B}_{i1}}$.

$$\frac{d}{dt}({}^c\overrightarrow{\mathbf{E}\mathbf{B}_{i1}}) = \varepsilon_i\frac{d}{dt}(-\mathbf{T}^c\underline{x}_b) = -\varepsilon_i\mathcal{K}\omega_z{}^c\underline{x}_b \quad (9)$$

where ω_z is the angular velocity of the end-effector around the fixed axis ${}^c\underline{\mathbf{z}}_e$ and ε_i denotes whether $\mathbf{B}_i$ is located on the same part as $\mathbf{E}$ or not (Fig. 4), namely $\varepsilon_1 = \varepsilon_2 = 1$ and $\varepsilon_3 = \varepsilon_4 = 0$.

Using (9), (8) can be written in matrix form:

$$\frac{d}{dt}({}^c\mathbf{B}_{i1}) = \begin{pmatrix} \mathbf{I}_3 & -\varepsilon_i\mathcal{K}^c\underline{x}_b \end{pmatrix}{}^c\mathcal{V}_e = \mathbf{G}_i{}^c\mathcal{V}_e \quad (10)$$

where

$$\mathbf{G}_i = \begin{pmatrix} \mathbf{I}_3 & -\varepsilon_i\mathcal{K}^c\underline{x}_b \end{pmatrix} \quad (11)$$

is the interaction matrix associated to the 3D moving point $\mathbf{B}_{i1}$ under the kinematic constraints and ${}^c\mathcal{V}_e = \begin{pmatrix} {}^cV_e^T & \omega_z \end{pmatrix}^T$ is the minimal representation of the end-effector Cartesian velocity.

Hence, by inserting (6) and (10) in (4), we obtain:

$$L^c\dot{\underline{\mathbf{u}}}_i = \begin{pmatrix} \mathbf{I}_3 & -\varepsilon_i\mathcal{K}^c\underline{x}_b \end{pmatrix}{}^c\mathcal{V}_e - l\dot{q}_i{}^c\underline{\mathbf{y}}_{pi} \quad (12)$$

Since $\underline{\mathbf{u}}_i$ is a unit vector (i.e. ${\underline{\mathbf{u}}_i}^T\dot{\underline{\mathbf{u}}}_i = 0$), we can obtain, from (12), the expression of the differential inverse kinematic model, associated to each joint velocities, of I4R:

$$\dot{q}_i = \frac{1}{l^c\underline{\mathbf{y}}_{pi}^T{}^c\underline{\mathbf{u}}_i}\begin{pmatrix} {}^c\underline{\mathbf{u}}_i^T & -\varepsilon_i\mathcal{K}^c\underline{\mathbf{u}}_i^T{}^c\underline{\mathbf{x}}_b \end{pmatrix}{}^c\mathcal{V}_e = {}^c\mathbf{D}_{ei}^{invc}\mathcal{V}_e \quad (13)$$

The differential inverse kinematic model associated to all joint velocities is thus:

$$\dot{\mathbf{q}} = {}^c\mathbf{D}_e^{invc}\mathcal{V}_e \quad (14)$$

where

$${}^c\mathbf{D}_e^{inv} = \begin{pmatrix} \frac{1}{l^c\underline{\mathbf{y}}_{p1}^T{}^c\underline{\mathbf{u}}_1} & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \frac{1}{l^c\underline{\mathbf{y}}_{p4}^T{}^c\underline{\mathbf{u}}_4} \end{pmatrix}\begin{pmatrix} {}^c\underline{\mathbf{u}}_1T & -\mathcal{K}^c\underline{\mathbf{u}}_1^T{}^c\underline{\mathbf{x}}_b \\ {}^c\underline{\mathbf{u}}_2^T & -\mathcal{K}^c\underline{\mathbf{u}}_2^T{}^c\underline{\mathbf{x}}_b \\ {}^c\underline{\mathbf{u}}_3^T & 0 \\ {}^c\underline{\mathbf{u}}_4^T & 0 \end{pmatrix} \quad (15)$$

To complete the differential inverse kinematic model [19] in order to explain all internal motions of the structure, we need to define the differential relation between ${}^c\dot{\underline{\mathbf{u}}}_i$ and ${}^c\mathcal{V}_e$ which gives the variation of the pointing direction of each leg in the camera frame:

$${}^c\dot{\underline{\mathbf{u}}}_i = \mathbf{M}_i{}^c\mathcal{V}_e \quad (16)$$

This relation will be needed for control and can be derived by inserting (13) in (12):

$$L^c\dot{\underline{\mathbf{u}}}_i = (\mathbf{I}_3 - \frac{{}^c\underline{\mathbf{y}}_{pi}{}^c\underline{\mathbf{u}}_i^T}{{}^c\underline{\mathbf{y}}_{pi}^T{}^c\underline{\mathbf{u}}_i})\begin{pmatrix} \mathbf{I}_3 & -\varepsilon_i\mathcal{K}^c\underline{\mathbf{x}}_b \end{pmatrix}{}^c\mathcal{V}_e \quad (17)$$

The differential inverse kinematic matrix associated to a leg orientation is hence:

$$\mathbf{M}_i = \frac{1}{L}(\mathbf{I}_3 - \frac{{}^c\underline{\mathbf{y}}_{pi}{}^c\underline{\mathbf{u}}_i^T}{{}^c\underline{\mathbf{y}}_{pi}{}^T{}^c\underline{\mathbf{u}}_i})\begin{pmatrix} \mathbf{I}_3 & -\varepsilon_i\mathcal{K}^c\underline{\mathbf{x}}_b \end{pmatrix} \quad (18)$$

IV. Vision Replaces Joint Sensors

The above differential inverse kinematic model depends on the following variables: the perpendicular $(\underline{\mathbf{y}}_{pi})$ to the arm (function of the joint values (q_i)) and the directions $(\underline{\mathbf{u}}_i)$ of each forearms. Consequently, one need to measure these variables. The first manner is to add redundant sensors, which is not always technically feasible. Using vision as a proprioceptive sensor we prove, in this paper, that these variables can be estimated only by vision. Moreover, it is shown that we can get rid of the joint values in the same move.

A. Variables Necessary to Vision-Based Control

In (14) and (16), one needs to compute ${}^c\underline{\mathbf{y}}_{p_i} = {}^c\underline{\mathbf{z}}_{p_i} \times {}^c\underline{\mathbf{x}}_{p_i}$ where ${}^c\underline{\mathbf{z}}_{p_i}$ is constant and ${}^c\underline{\mathbf{x}}_{p_i}$ depends on the joint values (q_i). However, noting that:

$${}^c\mathbf{B}_{i1} = {}^c\mathbf{A}_{i1} + L^c\underline{\mathbf{u}}_i = {}^c\mathbf{P}_i + {}^c\overrightarrow{\mathbf{P}_i\mathbf{A}_i} + {}^c\overrightarrow{\mathbf{A}_i\mathbf{A}_{i1}} + L^c\underline{\mathbf{u}}_i \quad (19)$$

we can express ${}^c\underline{\mathbf{x}}_{p_i}$ as:

$${}^c\underline{\mathbf{x}}_{pi} = \frac{1}{l}({}^c\mathbf{B}_{i1} - {}^c\mathbf{P}_i - {}^c\overrightarrow{\mathbf{A}_i\mathbf{A}_{i1}} - L^c\underline{\mathbf{u}}_i) \quad (20)$$

where l, L, ${}^c\mathbf{P}_i$ and ${}^c\overrightarrow{\mathbf{A}_i\mathbf{A}_{i1}}$ are constant parameters while the variable parameters ${}^c\underline{\mathbf{u}}_i$ can be measured by vision.

Consequently, the joint sensor value is not needed anymore provided that the variable ${}^c\mathbf{B}_{i1}$ can be estimated, which is the object of the next subsection.

B. Measurement Process

Recalling the assumption that the attachment point ${}^c\mathbf{B}_{i1}$ of the rod onto the travelling plate is lying on the revolution axis of the leg with radius R (Fig. 5), we can exploit the last constraint in (1) applied to both edges of the rod,

$$\begin{cases} {}^c\underline{\mathbf{n}}_1^{i\,T}\,{}^c\mathbf{B}_{i1} = -R \\ {}^c\underline{\mathbf{n}}_2^{i\,T}\,{}^c\mathbf{B}_{i2} = -R \end{cases} \tag{21}$$

Applying (21) to leg 1 and 2 yields:

$$\begin{cases} {}^c\underline{\mathbf{n}}_1^{1\,T}\,{}^c\mathbf{B}_{11} = -R \\ {}^c\underline{\mathbf{n}}_2^{1\,T}\,{}^c\mathbf{B}_{11} = -R \end{cases} \tag{22}$$

and

$$\begin{cases} {}^c\underline{\mathbf{n}}_1^{2\,T}\,{}^c\mathbf{B}_{21} = -R \\ {}^c\underline{\mathbf{n}}_2^{2\,T}\,{}^c\mathbf{B}_{21} = -R \end{cases} \tag{23}$$

Taking into account the travelling plate parameters (Fig. 4), one can have the following relation:

$${}^c\mathbf{B}_{11} = {}^c\mathbf{B}_{21} + \gamma^c\underline{x}_b - H^c\underline{z}_{p1} \tag{24}$$

Inserting (24) in (22), one gets:

$$\begin{cases} {}^c\underline{\mathbf{n}}_1^{1\,T}\,{}^c\mathbf{B}_{21} = -R - \gamma^c\underline{\mathbf{n}}_1^{1\,T}\,{}^c\underline{x}_b + H^c\underline{\mathbf{n}}_1^{1\,T}\,{}^c\underline{z}_{p1} \\ {}^c\underline{\mathbf{n}}_2^{1\,T}\,{}^c\mathbf{B}_{21} = -R - \gamma^c\underline{\mathbf{n}}_2^{1\,T}\,{}^c\underline{x}_b + H^c\underline{\mathbf{n}}_2^{1\,T}\,{}^c\underline{z}_{p1} \end{cases} \tag{25}$$

Finally, we obtain from (23) and (25) the following linear system from the image information:

$$\begin{pmatrix} {}^c\underline{\mathbf{n}}_1^{1\,T} \\ {}^c\underline{\mathbf{n}}_2^{1\,T} \\ {}^c\underline{\mathbf{n}}_1^{2\,T} \\ {}^c\underline{\mathbf{n}}_2^{2\,T} \end{pmatrix} {}^c\mathbf{B}_{21} = \begin{pmatrix} -R - \gamma^c\underline{\mathbf{n}}_1^{1\,T}\,{}^c\underline{x}_b + H^c\underline{\mathbf{n}}_1^{1\,T}\,{}^c\underline{z}_{p1} \\ -R - \gamma^c\underline{\mathbf{n}}_2^{1\,T}\,{}^c\underline{x}_b + H^c\underline{\mathbf{n}}_2^{1\,T}\,{}^c\underline{z}_{p1} \\ -R \\ -R \end{pmatrix} \tag{26}$$

The least-square solution ${}^c\mathbf{B}_{21}$ of this 4×3 linear system is unique provided that 3 of the interpretation planes are linearly independent. From (24), we can then compute ${}^c\mathbf{B}_{11}$.

Applying the same procedure on legs 3 and 4, we can build a second system for computing ${}^c\mathbf{B}_{31}$ and ${}^c\mathbf{B}_{41}$.

Let us point out that this estimation is performed in a single image. From this reconstruction we could estimate the joint value, but this additional step is not needed in the sequel.

V. VISUAL SERVOING

A. Edge Interaction Matrix

We propose to servo the error between the current ${}^p\underline{\mathbf{n}}_i^j$ and desired ${}^p\underline{\mathbf{n}}_i^{j\,*}$ cylinder edges . This control makes use of the detected edges in the image (Fig. 6).

To derive such a control, we need to express the interaction matrix ${}^p\mathbf{L}_i^j$ relating the Cartesian velocity ${}^c\mathcal{V}_e$ to the time derivative of the cylinder edges ${}^p\dot{\underline{\mathbf{n}}}_i^j$, expressed in the image frame:

$${}^p\dot{\underline{\mathbf{n}}}_i^j = {}^p\mathbf{L}_i^j\,{}^c\mathcal{V}_e \tag{27}$$

It can be computed as the product of two matrices:

$${}^p\mathbf{L}_i^j = {}^p\mathbf{J}_c\,{}^c\mathbf{L}_i^j \tag{28}$$

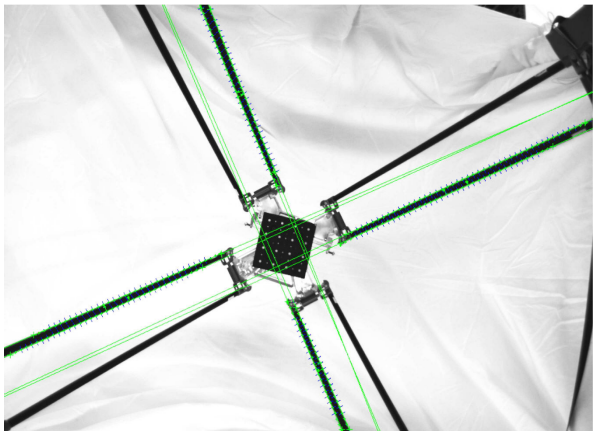

Fig. 6. Detected edges in the image.

where ${}^p\mathbf{J}_c$ is associated to the camera-to-pixel change of frame:

$${}^p\dot{\underline{\mathbf{n}}}_i^j = {}^p\mathbf{J}_c\,{}^c\dot{\underline{\mathbf{n}}}_i^j \tag{29}$$

and the second matrix ${}^c\mathbf{L}_i^j$ relates the time derivative of a cylinder edge ${}^c\dot{\underline{\mathbf{n}}}_i^j$, in the camera frame, to ${}^c\mathcal{V}_e$.

$${}^c\dot{\underline{\mathbf{n}}}_i^j = {}^c\mathbf{L}_i^j\,{}^c\mathcal{V}_e \tag{30}$$

B. Image Line Velocity and Edge Velocity

In [14], it was shown that image line velocity in pixel coordinates has the following expression:

$${}^p\mathbf{J}_c = \| \mathbf{K}^T\,{}^p\underline{\mathbf{n}}_i^j \| (\mathbf{I}_3 - {}^p\underline{\mathbf{n}}_i^j\,{}^p\underline{\mathbf{n}}_i^{j\,T})\mathbf{K}^{-T} \tag{31}$$

Following [14], by differentiation of constraints in (1) we can also express the edge velocity in the camera frame as:

$${}^c\dot{\underline{\mathbf{n}}}_i^j = (\mathbf{Q}_i^j\mathbf{G}_i + \mathbf{R}_i^j\mathbf{M}_i)\,{}^c\mathcal{V}_e = {}^c\mathbf{L}_i^j\,{}^c\mathcal{V}_e \tag{32}$$

with

$$\begin{cases} \mathbf{Q}_i^j = -\frac{({}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j)\,{}^c\underline{\mathbf{n}}_i^{j\,T}}{{}^c\mathbf{B}_{i1}{}^T\,{}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j} = -\frac{({}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j)\,{}^c\underline{\mathbf{n}}_i^{j\,T}}{{}^c\mathbf{B}_{i1}{}^T\,{}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j} \\ \mathbf{R}_i^j = -(\mathbf{I}_3 - \frac{({}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j)\,{}^c\mathbf{B}_{i1}{}^T}{{}^c\mathbf{B}_{i1}{}^T({}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{n}}_i^j)})\,{}^c\underline{\mathbf{u}}_i\,{}^c\underline{\mathbf{n}}_i^{j\,T} \\ {}^c\mathbf{L}_i^j = \mathbf{Q}_i^j\mathbf{G}_i + \mathbf{R}_i^j\mathbf{M}_i \end{cases} \tag{33}$$

where $\mathbf{G}_i$ is defined in (11) and $\mathbf{M}_i$ in (18).

C. Control

Control is performed along the classical visual servoing scheme. The individual error grounding the control law is the geodesic error:

$$\mathbf{e}_{i,j} = {}^p\underline{\mathbf{n}}_i^j \times {}^p\underline{\mathbf{n}}_i^{j\,*} \tag{34}$$

whose time derivatives are

$$\dot{\mathbf{e}}_{i,j} = {}^p\dot{\underline{\mathbf{n}}}_i^j \times {}^p\underline{\mathbf{n}}_i^{j\,*} = -[{}^p\underline{\mathbf{n}}_i^{j\,*}]_\times\,{}^p\dot{\underline{\mathbf{n}}}_i^j \tag{35}$$

Introducing:

$$\mathbf{N}_i^j = -[{}^p\underline{\mathbf{n}}_i^{j\,*}]_\times\,{}^p\mathbf{L}_i^j \tag{36}$$

equations (27) and (36) give:

$$\dot{\mathbf{e}}_{i,j} = \mathbf{N}_i^j {}^c\mathcal{V}_e \tag{37}$$

This yields the following pseudo-control vector ${}^c\mathcal{V}_e$:

$${}^c\mathcal{V}_e = -\lambda \mathbf{N}^+ \mathbf{e} \tag{38}$$

where $\mathbf{N}$ is the compound matrix from the associated individual interaction matrices $\mathbf{N}_i^j$.

Inserting (38) into (14) delivers the final control law:

$$\dot{\mathbf{q}} = -\lambda \widehat{{}^c\mathbf{D}_e^{inv}} \widehat{\mathbf{N}^+} \mathbf{e} \tag{39}$$

Notice that this control is performed directly on the image signal and that the reconstructed $\mathbf{B}_{i1}$ and $\underline{\mathbf{u}}_i$, as well as all the other parameters (camera intrinsic parameters and kinematic parameters) only appear in the interaction matrix. Hence, estimation errors on these variables and parameters do not jeopardize the convergence provided that the usual definitive positive condition of the interaction matrix is ensured.

VI. EXPERIMENTAL RESULTS

A. Attachment Points Estimation

In a first experiment, we applied the measurement process (26) to a perspective camera placed vertically between the legs and fixed to the base (${}^c\underline{\mathbf{z}}_c \approx -{}^b\underline{\mathbf{z}}_b$). Line extraction is performed using ViSP [20] an open C++ library for visual servoing.

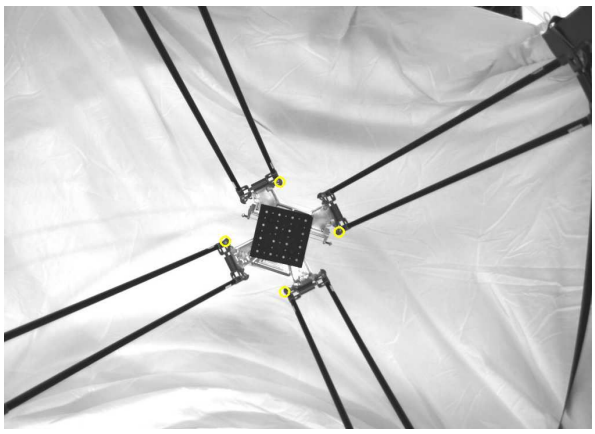

Fig. 7. Re-projection of the joint locations ${}^c\mathbf{B}_{i1}$ in the image.

A qualitative evaluation of the experiment is to project the estimated attachment points ${}^c\mathbf{B}_{i1}$ onto the image plane. Fig. 7 shows that the estimation is correct in the horizontal plane (yellow circles).

B. Robustness to Noise

In the simulations presented below, we choose for initial position ${}^b\mathbf{E} = (0, 0, -950mm)^T$ as the end-effector position and $\theta = 0°$.

Then, the desired position (Fig. 8) is obtained from the reference position by a translation along the 3 axes and a rotation around the $\underline{\mathbf{z}}_e$ axis (${}^b\mathbf{E} = (150mm, 50mm, -1000mm)^T$ and $\theta = 20°$). Fig. 9 shows that the errors on each leg converge exponentially to 0, with

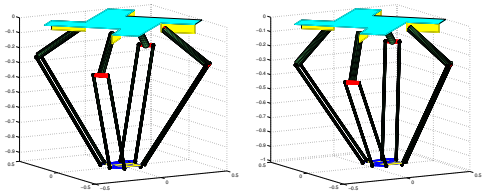

Fig. 8. Desired (right) and initial (left) position, case of the I4R prototype.

a perfect decoupling. It shows also that the Cartesian error converges to 0. We verified that, in the reported simulation, the robot reaches the desired position from the initial one.

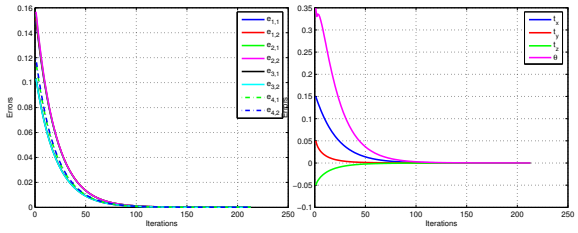

Fig. 9. Evolution of the (unit-less) error $e_{i,j}^T e_{i,j}$ on each edge (left) and evolution of the Cartesian error (right), case of the I4R prototype.

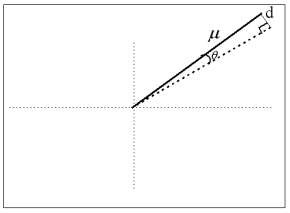

Fig. 10. Effect of the image edge rotation on the distance covered by its extremities.

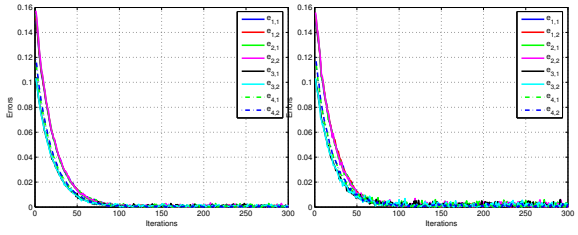

Fig. 11. Robustness to noise: Evolution of the (unit-less) error $e_{i,j}^T e_{i,j}$ on each edge, with a noise amplitude of $0.2°$ (left) and $0.4°$ (right), case of the I4R prototype.

To show the robustness of the approach, we added noise to the detected edges in the image expressed in the camera frame. Assuming that an image edge is essentially a unit vector, the image noise can be modelled by a rotation of the unit vector of each edge. Consequently, we define at each time a sample random rotation axis and a positive rotation angle with maximal amplitude of $0.2°$ and $0.4°$. It has to be noticed, for a quantitative perception of such an error model,

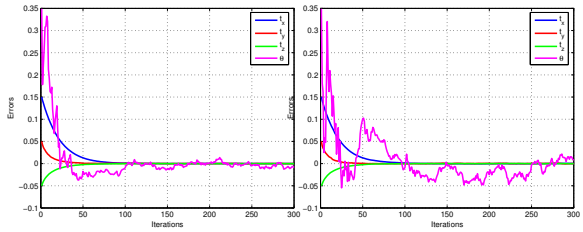

Fig. 12. Robustness to noise: Evolution of the cartesian error, with a noise amplitude of $0.2°$ (left) and $0.4°$ (right), case of the I4R prototype.

	Noise amplitude of $0.2°$	Noise amplitude of $0.4°$
Iteration	>100	>100
t_x (mm)	± 0	± 0.235
t_y (mm)	± 0.298	± 0.698
t_z (mm)	± 0	± 0.01
θ(deg)	± 0.779	± 2.75

TABLE II

3D ERROR AMPLITUDE DURING THE CONVERGENCE TAILS IN FIG. 12

that a rotation angle of $0.2°$ around the axis perpendicular to both $\underline{\mathbf{u}}_i$ and $\underline{\mathbf{n}}_i^j$ (Fig. 10) yields an error ($d = \mu\theta$) of about 2 pixels on the extremities of a 600 pixel-long line segment (μ) which is fairly what one gets with a line tracker. Fig. 11 shows a good robustness, in the image, of the control with respect to such a noise.

Fig. 12 and Table II show a good accuracy of this control in the translation of the end-effector but a high sensitivity to noise of the end-effector orientation. However, this sensitivity is due to the high amplification gain of the relative translation of the nacelle.

VII. CONCLUSIONS AND FUTURE WORKS

In this paper, we have introduced an image-based visual servoing of the I4R robot without proprioceptive sensors. This has induced the development of a measurement process to estimate the attachment points of the legs onto the travelling plate.

The main advantages of this method are that the forward kinematic model is never used in the control and that the kinematics parameters and reconstructed variables ($\mathbf{B}_{i1}$ and $\underline{\mathbf{u}}_i$) only appear in interaction or Jacobian matrices. This yields a higher robustness to calibration errors since the control is done directly in the image space. Then, we do not require the use of a visual pattern to estimate the relative pose of the later with respect to the end-effector. This approach can hence be used when the end-effector is not visible.

Taking this approach, we confirmed that, measuring by vision the leg edges, one can easily measure a projective kinematic model for control. No proprioceptive sensor are required. In the next work, we would like to experiment on a real prototype and to show if accuracy in rotation increases by taking into account both rods of the forearms.

REFERENCES

[1] L. Weiss, A. Sanderson and C. Neuman. Dynamic sensor-based control of robots with visual feedback. *IEEE Trans. on Robotics and Automation*, 3(5), October 1987.

[2] 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.

[3] S. Hutchinson, G. Hager and P. Coke. A tutorial on visual servo control. *IEEE Trans. on Robotics and Automation*, 12(5), pages 651 – 670, October 1996.

[4] 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.

[5] H. Kino, C.C. Cheah, S. Yabe, S. Kawamua 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.

[6] 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.

[7] T. Dallej, N. Andreff and P. Martinet. 3D Pose Visual Servoing Relieves Parallel Robot Control from Joint Sensing. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'06*, pages 4291-4296, Beijing, China, October 9-15 2006.

[8] B. Thuilot, P. Martinet, L. Cordesses and J. Gallice. Position based visual servoing: keeping the object in the field of vision. *Proceedings of the IEEE International Conference on Robotics and Automation* (ICRA'02), pages 1624–1629, May 2002.

[9] 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.

[10] 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.

[11] N. Andreff, A. Marchadier and P. Martinet. Vision-based control of Gough-Stewart parallel mechanism using legs observation. In *Int. Conf. Robotics and Automation (ICRA'05)*, pages 2546-2551, Barcelona, Spain, May 2005.

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

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

[14] N. Andreff, T. Dallej and P. Martinet. Image-based visual servoing of Gough-Stewart parallel manipulators using legs observation. In *Proceedings of the 8th International IFAC Symposium on Robot Control SYROCO 2006*, Santa Cristina Convent, University of Bologna (Italy) September 6 - 8, 2006.

[15] S. Krut, V. Nabat, O. Company and F. Pierrot. A high speed robot for SCARA motions. In *Proc. IEEE/ICRA International Conference on Robotics and Automation, ICRA'04*, pages 4109–4115, New Orleans, USA, April 26 - May 1 2004.

[16] F. Pierrot and O. Company. H4: A new family of 4-dof parallel robots. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM'99*, pages 508–513, Atlanta, Georgia, USA, September 1999.

[17] S. Krut, O. Company, M. Benoit, H. Ota and F. Pierrot. I4: A new parallel mechanism for SCARA motions. In *Proc. IEEE/ICRA International Conference on Robotics and Automation, ICRA'03*, pages 1875–1880, Taipei, Taiwan, September 2003.

[18] V. Nabat, M. O. Rodrigues, O. Company, S. Krut and F. Pierrot. Par4: very high speed parallel robot for pick-and-place. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'05*, pages 1202–1207, Alberta, Canada, August 2-6 2005.

[19] N. Andreff and P. Martinet. Kinematic modelling of some parallel manipulators for vision-based control purposes. In *Proc. of EuCoMes, the first European Conference on Mechanism Science*, Austria, February 21-26 2006.

[20] E Marchand, F Spindler and F Chaumette. ViSP for visual servoing: a generic software platform with a wide class of robot control skills. *IEEE Robotics and Automation Magazine, Special Issue on "Software Packages for Vision-Based Control of Motion", P. Oh, D. Burschka (Eds.)*, pages 40–52, 12(4), December 2005.