One Camera in Hand for Kinematic Calibration of a Parallel Robot

Alberto Traslosheros, José María Sebastián, Eduardo Castillo, Flavio Roberti, Ricardo Carelli

Abstract—The main purpose of robot calibration is the correction of the possible errors in the robot parameters. This paper presents a method for a kinematic calibration of a parallel robot that is equipped with one camera in hand. In order to preserve the mechanical configuration of the robot, the camera is utilized to acquire incremental positions of the end effector from a spherical object that is fixed in the word reference frame. Incremental positions of the end effector are related to incremental positions of encoders of the motors of the robot. A kinematic model of the robot is modified in order to take into account possible errors of kinematic parameters. The solution of the model utilizes incremental positions of the resolvers and end effector, the new parameters minimizes errors in the kinematic equations. Spherical properties and intrinsic camera parameters are utilized to model sphere projection in order to improve spatial measurements. The robot system is designed to carry out tracking tasks and the calibration of the system is finally validated by means of integrating the errors of the visual controller.

I. INTRODUCTION

ACCURACY is a crucially important performance specification for parallel kinematic machines (PKM), particularly for those that are involved in tasks as can be surgery, material machining, electronics manufactory, products assembling, among other. In this work calibration method is used to improve a widely studied image based visual controller [1]. In contrast to serial robots, parallel robots (or parallel kinematic machines) are characterized by high structural stiffness, high load operation, high speed and acceleration of the end effector and, high accuracy for end effector positioning. This accuracy, however, relies on a robust and accurate calibration, which is a difficult problem both from a theoretical and a practical point of view, even if it may be performed off-line. Robot accuracy can be affected by increasing backlash due to robot operation, thermal effects, elements deformation [2], robot control and manufacturing errors. Although hybrid calibration methods exist, in general, it is possible to classify different calibration strategies in three main groups (by considering the location of measurement instruments and its additional elements): External calibration, it is based on measurements of the positions of the end effector (or other structural element) of the robot by means of an external instrument. Constrained calibration groups those methods that rely on mechanical elements in order to constraint some kind of motion of the robot during the calibration process, this method is generally simple and it is considered the most inexpensive. And finally, auto calibration that consist on that methods that calibrates the robot automatically and even during the robot operation [3], in general auto (or self) calibration method is more expensive due to its complexity and even it includes redundant sensors [4] or elements. External calibration can be done by measuring completely or partially the pose parameters of the platform. Measurements of the pose of a platform can be done with a laser and a coordinate measuring machine (CMM)[5], commercial visual systems (optical system and infrared light) [6], laser sensor [7], by adding passive legs [8], with a interferometer [9], by a LVDT and inclinometers in [10] [11], theodolite [12], with gauges [13], with double ball bar (DBB) [14], by inspecting a machined part that is dedicated to the calibration process [15], accelerometers [16], or with visual systems and patterns widely studied (chess-board) [17][18][19]. Above examples are different strategies that obtain kinematic information of the robot but in general, calibration methods impose virtual or real constraints on the poses of the end effector (or mobile elements). By choosing the appropriate method, a calibration can be an economical and practical technique for improve accuracy of a PKM [20].

The principle of the calibration process is to get the constraints at a large enough number of measured poses (called calibration poses) in order to conclude which is the best geometry (distances or angles of elements) of the robot that satisfies them. The basic idea it has been applied to calibration of serial and parallel robots and the main difference has been the measurement instruments and strategies. Visual methods are becoming more popular due to its simplicity and because it can be an inexpensive method in comparison to others, for parallel robots it was first proposed by Amirat [21]. Visual methods that propose a monocular system [17][18][19] propose to utilize a pattern (marks on a flat surface), these kind of patterns are employed in camera calibration where it is possible to obtain intrinsic and extrinsic parameters [22]. By means of the extrinsic parameters of the camera it is possible (by attaching the camera or the pattern to the rend effector of the robot) to obtain the pose of the end effector of the robot and

A. Traslosheros and J. M. Sebastián are with the Technical University of Madrid, DISAM, José Gutierrez Abascal 2, CO 28003 Spain. Phone (+34) 913 363 061, email: atraslosheros@etsii.upm.es and jsebas@etsii.upm.es. E. Castillo is with the National Technical Institute of Mexico, Departament of Mechatronics, Cerro Blanco No. 141., CO 76090, Queretaro, Mexico. Phone (+52) 442 229 0804, email: ecastilloca@ipn.mx

F. Roberti and R. Carelli are with the National University of San Juan, Av. San Martín Oeste 1109, CO 5400 Argentina. Phone (+54) 0264 4 213 303, email: froberti@inaut.unsj.edu.ar and rcarelli@inaut.unsj.edu.ar. Department of Automatics, National University of San Juan, Argentina.

consequently those poses can be considered as calibration poses.

In this paper an economical and external calibration method is proposed to calibrate a parallel robot of 3 DOF. The proposed method utilizes positions from the resolvers of the motors and visual information that is obtained from a spherical element (in order to obtain joints and 3D incremental positions). Once the above information is obtained then it is replaced in the constraint equations in order to solve numerically and obtain the best set of geometrical parameters that satisfies the set of equations. Finally calibrated parameters are used in an image visual servoing controller and compared to the controller that utilizes nominal parameters.

II. System Description

The main objective of proposed method it is to calibrate the parallel robot of the system called: Robotenis. Calibration of the robot is required in order to improve its accuracy in visual servoing algorithms and develop future tasks as can be the Ping-Pong game. Robotenis was designed in order to study and design visual servoing controllers and to carry out object tracking under dynamic environments. The mechanical structure of system Robotenis is inspired by the DELTA robot [23][24] and its vision system is based in one camera allocated at the end effector of the robot. Basically, the platform Robotenis (Fig. 1 a) consists of a parallel robot and a visual system for acquisition and analysis. The maximum end effector speed of the parallel robot is $6\,m/s$ and its maximum acceleration is $58\,m/s^2$. The visual system of the platform Robotenis [25] is composed of a 50 grams camera that is located at the end effector (Fig. 1 b) and a frame grabber (SONY XCHR50 and Matrox Meteor 2-MC/4 respectively). The motion system is composed of three AC brushless servomotors, Ac drivers (Unidrive), planetary gearboxes and the joint controller is implemented in a DS1103 card (implemented in ANSI C).
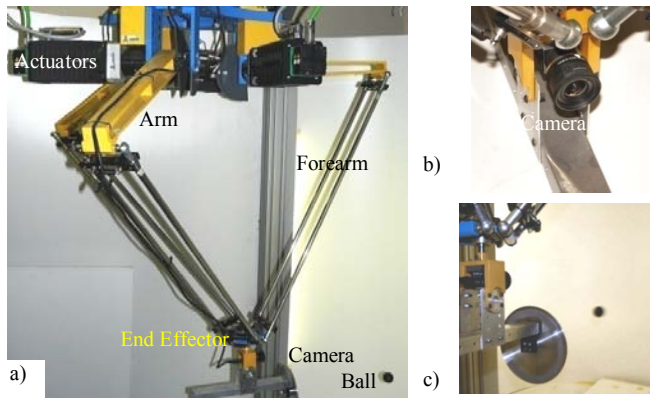

Fig. 1 a) System Robotenis, b) Robot camera, c) Robot environment

III. Constraint Equations

Constraint equations of the calibration method were obtained from a kinematic model of the robot. It is important

to bring out that in the constraint equations, transformation matrix from the camera to the end effector is not considered.

Transformation matrix was divided in its rotation and translation, rotation matrix was obtained independently to the robot calibration, on the other hand translation matrix is not necessary, because the camera is fixed to the end effector and the model of the constraint equations is an incremental one. In the Fig. 2 it is shown a sketch of the parallel robot, where O_{xyz} is the robot reference frame, a_i are arm lengths, b_i are forearm lengths, arms and forearms are connected in B_i, H_i are distances from the robot reference frame (O_{xyz}) to the revolute axis of motors (A_i), h_i are the distances from the center of the end effector (P) to the forearm (C_i), θ_i are the motor angles from a home position (unknown), ϕ_i is the angle in which point A_i is allocated in the plane XY of the robot frame reference and $i = 1,2,3$.

Constraint equations are based on the movement of the forearms respect to the robot arms. [26]. In the Delta robot, forearms are parallelograms that link the end effector to the arms by means of spherical joints. Spherical joints allow modeling the movement of the end effector as the intersection of three spherical surfaces that are described by points C_i in Fig. 3. By inspecting Fig. 2 and Fig. 3 it is possible to obtain the surface of a sphere described by the point C_i with center in B_i as:

$$\Gamma_i = (C_{ix} - B_{ix})^2 + (C_{iy} - B_{iy})^2 + (C_{iz} - B_{iz})^2 - b_i^2 = 0 \qquad (1)$$

Where B_i and C_i can be expressed in the robot frame as:

$$^{0_{xyz}}\begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix}_i = \begin{bmatrix} (H + a\,c\theta)\,c\phi \\ (H + a\,c\theta)\,s\phi \\ a\,s\theta \end{bmatrix}_i$$

$$^{0_{xyz}}\begin{bmatrix} C_x \\ C_y \\ C_z \end{bmatrix}_i = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix}_i - \begin{bmatrix} c\phi & -s\phi & 0 \\ s\phi & c\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}_i \begin{bmatrix} -h \\ 0 \\ 0 \end{bmatrix}_i$$

$$(2)$$

Note that in above equations $c\theta_i = \cos\theta_i$, $s\theta_i = \sin\theta_i$, $P = [P_x, P_y, P_z]'$, $B_i = [B_x, B_y, B_z]'_i$, etc. By other hand by supposing that we can obtain incremental joint and end effector positions we have that each absolute measurement can be expressed as:

$$\theta_{ik} = \theta_{i0} + \Theta_{ik} \quad \text{and} \quad P_k = P_0 + \Pi_k \qquad (3)$$

Observe that θ_{i0} and P_0 are unknown and, $\theta_{ik} = \Delta\theta_{ik} = \theta_{ik} - \theta_{i(k-1)}$ and $\Pi_k = \Delta P_k = P_k - P_{k-1}$ are incremental measuring (obtained from resolvers and camera): $V_k = [\Theta_1, \Theta_2, \Theta_3, \Pi_x, \Pi_y, \Pi_z]_k$, where Π_k are measured and are parameters that are expressed as incremental positions of the end effector and they are measured from sphere images (section V), k is the incremental measurement number.

IV. Solution of the Constraints Equations

Not all parameters of the robot have effect on the kinematic model and as a consequence they cannot be identified, this is known as observability. An observability measure can be derived from the Jacobian of the constraint equations. This

Jacobian is called the observation matrix and observable parameters can be obtained by its **QR** decomposition [27]. In our model by means of the **QR** decomposition influence of the parameters ϕ_i and H_i are difficult to observe in the model, for this reason they are not identified in this work and them nominal value is taken into account. On the other hand identifiable parameters are: a_i , b_i, h_i and reference position of the actuators and the end effector of the robot: θ_{i0} and $\boldsymbol{P_0}$ respectively ($i = 1,2,3$ and $\boldsymbol{P_0}$ is the initial end effector position). Thus unknown parameters can be expressed as: $\boldsymbol{u_i} = [a_i , b_i , h_i , \theta_{i0}]$ and $\boldsymbol{U} = [u_1\ u_2\ u_3\ P_{x0}\ P_{y0}\ P_{z0}]$

And substituting incremental equations (3) in (2) and (1), constraint equations can be arranged as:

$$\Gamma_i = \left((a_i\, c(\theta_i + \Theta_{ik}) + H_i)\, c\phi_i - P_x - h_i\, c\phi_i - \Pi_{xk}\right)^2$$
$$+ \left((a_i\, c(\theta_i + \Theta_{ik}) + H_i)\, s\phi_i - P_y - h_i\, s\phi_i - \Pi_{yk}\right)^2 \quad (4)$$
$$+ (a_i\, s(\theta_i + \Theta_{ik}) - P_z - \Pi_{zk})^2 - b_i^2 = 0$$

In order to solve, equations in (4) are grouped as:

$$\Phi(U, V_k) = [\Gamma_{11}, \dots, \Gamma_{1k}, \Gamma_{21}, \dots, \Gamma_{2k}, \Gamma_{31}, \dots, \Gamma_{3k}]' \quad (5)$$

Note that constraint equations are not linear, its solution is no exactly satisfied thus commonly a nearly solution is obtained by numerical algorithms as can be Gauss-Newton. Expressing constraints (5) in them Taylor linear approximation in order to solve iteratively.

$$\Phi(U_{n+1}, V_k) \approx \Phi(U_n, V_k) + J(U_n, V_k)(U_{n+1} - U_n) = 0 \quad (6)$$

Where n is the iteration number and $J_{3k \times 18}$ is the Jacobian of the constraint equations or the observation matrix that it is given by:

$$J(U_n, V_k) = \begin{bmatrix} J_{1k} \\ J_{2k} \\ J_{3k} \end{bmatrix}$$

$$J_{1k} = \begin{bmatrix} \frac{\partial \Gamma_{11}}{\partial a_1} \cdots \frac{\partial \Gamma_{11}}{\partial \theta_{10}}, 0, \dots, 0, 0, \dots, 0, \frac{\partial \Gamma_{11}}{\partial P_{x0}}, \frac{\partial \Gamma_{11}}{\partial P_{y0}}, \frac{\partial \Gamma_{11}}{\partial P_{z0}} \\ \vdots \\ \frac{\partial \Gamma_{1k}}{\partial a_1} \cdots \frac{\partial \Gamma_{1k}}{\partial \theta_{10}}, 0, \dots, 0, 0, \dots, 0, \frac{\partial \Gamma_{1k}}{\partial P_{x0}}, \frac{\partial \Gamma_{1k}}{\partial P_{y0}}, \frac{\partial \Gamma_{1k}}{\partial P_{z0}} \end{bmatrix}$$

$$J_{2k} = \begin{bmatrix} 0, \dots, 0, \frac{\partial \Gamma_{21}}{\partial a_2} \cdots \frac{\partial \Gamma_{21}}{\partial \theta_{20}}, 0, \dots, 0, \frac{\partial \Gamma_{21}}{\partial P_{x0}}, \frac{\partial \Gamma_{21}}{\partial P_{y0}}, \frac{\partial \Gamma_{21}}{\partial P_{z0}} \\ \vdots \\ 0, \dots, 0, \frac{\partial \Gamma_{2k}}{\partial a_2} \cdots \frac{\partial \Gamma_{2k}}{\partial \theta_{20}}, 0, \dots, 0, \frac{\partial \Gamma_{2k}}{\partial P_{x0}}, \frac{\partial \Gamma_{2k}}{\partial P_{y0}}, \frac{\partial \Gamma_{2k}}{\partial P_{z0}} \end{bmatrix} \quad (7)$$

$$J_{3k} = \begin{bmatrix} 0, \dots, 0, 0, \dots, 0, \frac{\partial \Gamma_{31}}{\partial a_3} \cdots \frac{\partial \Gamma_{31}}{\partial \theta_{30}}, \frac{\partial \Gamma_{31}}{\partial P_{x0}}, \frac{\partial \Gamma_{31}}{\partial P_{y0}}, \frac{\partial \Gamma_{31}}{\partial P_{z0}} \\ \vdots \\ 0, \dots, 0, 0, \dots, 0, \frac{\partial \Gamma_{3k}}{\partial a_3} \cdots \frac{\partial \Gamma_{3k}}{\partial \theta_{30}}, \frac{\partial \Gamma_{3k}}{\partial P_{x0}}, \frac{\partial \Gamma_{3k}}{\partial P_{y0}}, \frac{\partial \Gamma_{3k}}{\partial P_{z0}} \end{bmatrix}$$

Finally parameters are iteratively obtained by:

$$U_{n+1} = U_n - J(U_n, V_k)^+ \Phi(U_n, V_k) \quad (8)$$

Where J^+ is the Jacobian pseudoinverse. Initial values (U_0) are given by the nominal parameters and a pose of the robot (given from nominal parameters). In the constraint equations in (4), units of the Jacobian (7) are homogeneous and Jacobian is not necessary to normalize
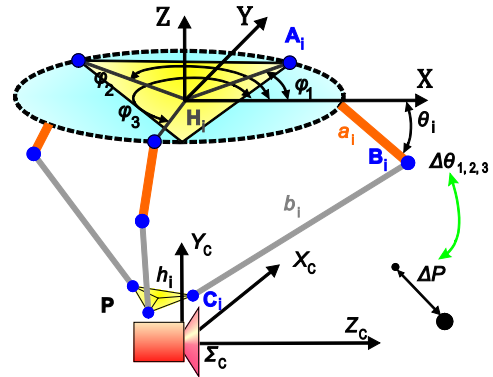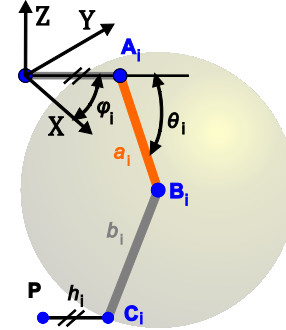

Fig. 2 Sketch of the system Robotenis.

Fig. 3 One leg profile of the system Robotenis.

V. VISUAL MEASUREMENT SYSTEM

Visual measurement system is composed by a fixed ball (hanging by a thread) and a calibrated camera that it is allocated on the end effector of the robot. Initial positions of resolvers are registered; incremental positions are measured by the visual system and both measures related by constraint equations to solve unknown parameters.

A. Selection of Calibration Poses

In order to solve the constraint equations system a set of incremental (resolvers and end effector) poses is needed, simulations shows that 6 different poses are at least required (perfect measurements). However perfect measurements do not exist and is necessary to choose best calibration poses. Thus, errors in the parameters of the robot are especially critical when the calibration pose is near from a singularity. However in a practical calibration process, if poses are extremely near of singularities, those poses are able to produce unpredictable movements (due to kinematic errors). In this work, the robot workspace is numerically well known (by using nominal values) and the calibration poses are chosen randomly from the workspace boundaries. Thus if a pose is between 5 and 10 cm away from a singularity, then the pose is included as a calibration pose. Experiments were implemented with 300 of poses and each measure is the mean of 500 images in order to avoid noise from image acquisition (image acquisition rate: $8.34\ ms$). Due to its simplicity, calibration process is done automatically in 21 minutes and it is possible to calibrate on-line.

On the other hand, a spherical object has been chosen as reference of the end effector and some considerations of the

spherical object and its projection on the image plane has to be taken into account, Fig. 4. Each measurement is done in two iterative steps, first a simplified model is considered as in Fig. 5 where two sub-pixel points are acquired from the image (distortion is previously corrected), secondly a correction of measurements is done by considering elliptical projection of the ball.

B. Simplified Ball Projection Model

Objective of this method is to obtain the position of the center of a spherical object. Position is obtained from the projection of the ball in the plane of the image. The method takes into account that the projected ball does not correspond to the real center and diameter of the ball. Proposed method is iterative and a first approximation is obtained by a simplified 2D model as shown in fig. 5. Model in fig. 5 is contained in a plane in the coordinates X and Z that are in the camera coordinate frame (X_c, Y_c, Z_c). On the other hand points that are in the image are related by a line that is tangent to the sphere and passes through optical center of the camera, that is:

$$X = \alpha_1 Z \qquad \text{and} \qquad X = \alpha_2 Z \qquad (9)$$

Sphere is fixed in the space, its diameter is known (38mm) and the distance of the line that is tangent to the perimeter to the center of the sphere (L_B, C_B) are related by:

$$R_B = \frac{L_B - \alpha C_B}{\sqrt{1 + \alpha^2}} \qquad (10)$$

Where $R_B = 19mm$ and α can be cleared from eq.(10):

$$\alpha_{1,2} = \frac{L_B C_B \pm \sqrt{L_B^2 C_B^2 - (C_B^2 - R_B^2)(L_B^2 - R_B^2)}}{C_B^2 - R_B^2} \qquad (11)$$

Where C_B is supposed positive and $C_B > R_B$, otherwise the ball cannot be seen. By substituting eq. (10) and (9) in the following circumference equation from Fig. 5:

$$(\alpha_{1,2} Z - L_B)^2 + (Z - C_B)^2 = R_B^2 \qquad (12)$$

Clearing for Z and simplifying it is possible to obtain:

$$Z_{1,2} = \frac{\alpha_{1,2} L_B + C_B}{\alpha_{1,2}^2 + 1} \qquad (13)$$

Thus the purpose is to obtain the center of the ball (L_B, C_B) from the known points in the image (x_{u1}, x_{u2}). For points that are tangent to the sphere it is easy to see that:

$$\frac{X_1}{Z_1} = \frac{x_{u1}}{f} \quad ; \quad \frac{X_2}{Z_2} = \frac{x_{u2}}{f} \qquad (14)$$

Where f is the focal distance. The algorithm that is implemented is iterative and its solution is obtained in a few iterations. Initial solution is obtained supposing that:

$$Z_1 \approx Z_2 \approx C_b \quad ; \quad |X_2 - X_1| = 2R_B \qquad (15)$$

In equation (11) parameters L_B and C_B are the position of the center ($X_B = L_B$ and $Z_B = C_B$) of the circle in Fig. 5 and they are unknown. At this point, must be cleared that the algorithm is iterative and it needs an initial position that is given by:

$$C_B = \frac{2fR_B}{|x_{u2} - x_{u1}|} \quad \text{and} \quad L_B = \frac{x_{u2} + x_{u1}}{2f} C_B \qquad (16)$$

Values obtained in eq. (16) and equations (11), (13) and (14) are iteratively utilized to obtain C_B and L_B that are repeatedly substituted in eq. (12) until there are not differences between old and new values.
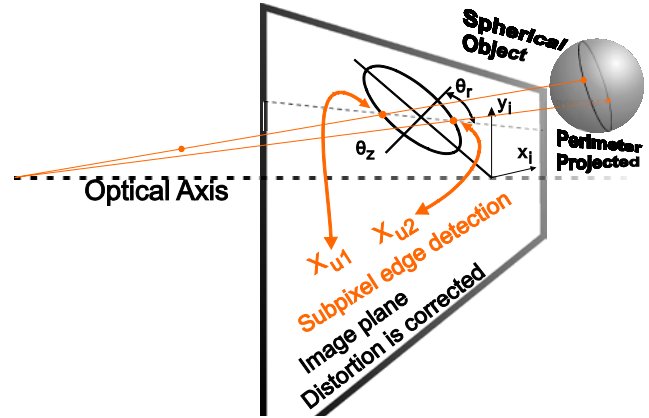

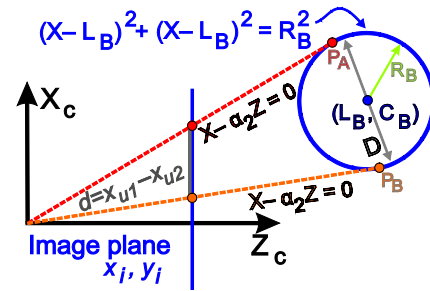Fig. 4 Ball projection in the image plane.

Fig. 5 Ball projection in the image plane.

A. Ellipse Ball Projection in Z correction

One condition of the method is that the spherical object must be fully visible. This condition guarantees that the projection of the sphere is an ellipse Fig. 6 (or a circumference if the projection lies on the center of the image). Note in Fig. 6 that x_{u1} and x_{u2} are not on the axis of the projected ellipse, thus the diameter considered in the calculus of Z contains a small error due to the projected ball.

In order to correct the diameter that is considered in the calculus of Z, the minor axis of the ellipse has to be calculated. Once the minor axis is calculated then $2b_e$ is replaced in (16) instead of $|x_{u2} - x_{u1}|$, thus the ball position is calculated by earlier method for a second time. This second step mainly depends on angle θ_z and the method stops when θ_z is constant or almost constant (usually two or three cycles). θ_z is the angle between the optical axis and the line that passes through the center of the ball and the center of the projected ball. Note that θ_z coincides with the angle that the image plane can be rotated around the minor axis of the ellipse in order to project a circle on the plane image, thus:

$$\theta_z = \text{acos}\left(\frac{Z}{\sqrt{(X^2 + Y^2 + Z^2)}}\right) \qquad (17)$$

The minor (b_e) and major (a_e) axes of the projected

ellipse are related by:

$$b_e = a_e \cos(\theta_z) \tag{18}$$

The projected ellipse in its rotated canonical parametric form can be expressed as:

$$x_{ue} = b_e \cos(\theta_e) \cos(\theta_r) - a_e \sin(\theta_e) \sin(\theta_r) + x_{uc}$$
$$y_{ue} = b_e \cos(\theta_e) \sin(\theta_r) + a_e \sin(\theta_e) \cos(\theta_r) + y_{uc} \tag{19}$$

Where θ_e is the angle of a point in the canonical form of the ellipse, θ_r is the angle that the ellipse is rotated around the Z axis of the camera (eq. (20)) and (x_{uc}, y_{uc}) is the center of the ellipse in the image, Fig. 7.

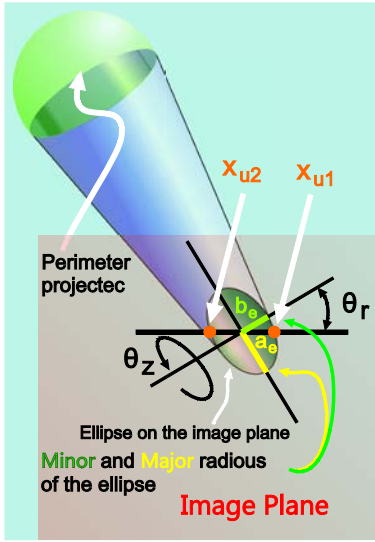$$\theta_r = \text{atan}\left(-\frac{y_{uc}}{x_{uc}}\right) \tag{20}$$

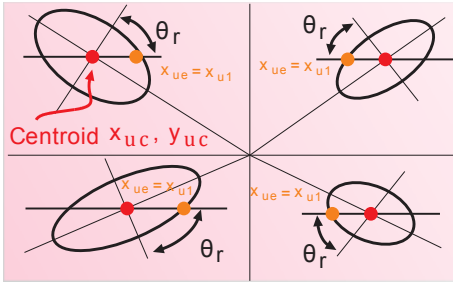Fig. 6 Ellipse of the sphere projection on the image plane.

Fig. 7 Angle θ_r.

On the other hand we know that (x_{u1}, y_{u1}) and (x_{u2}, y_{u2}) belong to the ellipse, and by simple inspection in Fig. 6:
$x_{uc} - x_{ue} = (x_{u2} - x_{u1})/2$, $y_{uc} - y_{ue} = (y_{u2} - y_{u1})/2 = 0$ and substituting in (19) we can obtain that:

$$(x_{u2} - x_{u1})/2 = b_e \cos(\theta_e) \cos(\theta_r) - a_e \sin(\theta_e) \sin(\theta_r)$$
$$0 = b_e \cos(\theta_e) \sin(\theta_r) + a_e \sin(\theta_e) \cos(\theta_r) \tag{21}$$

Substituting (18) in (21) and clearing for a_e and θ_e finally:

$$a_e = \left| \frac{(x_{u2} - x_{u1})/2}{\cos(\theta_z) \cos(\theta_e) \cos(\theta_r) - \sin(\theta_e) \sin(\theta_r)} \right| \tag{22}$$

$$\theta_e = \text{atan}(-\tan(\theta_r) \cos(\theta_z))$$

VI. RESULTS

In order to test the calibration method, nominal parameters and calibrated parameters are compared when they are utilized in an image based visual controller. The robot and its visual controller are designed with the purpose of carry out tracking tasks, for more information please consults [28]. Visual controller is based in a well established architecture called: dynamic image-based look-and-move visual servoing. In this scheme visual controller makes use of image estimates (position, velocity), Jacobian of the robot and its kinematic model in order to act over the robot actuators. Thus errors in the kinematic model generate undesirable movements that have an effect on the performance of the visual controller. In this section we compare the performance of the visual controller of the robot when the controller makes use of the nominal kinematic parameters and calibrated kinematic parameters (shown in table 1). In order to compare the visual controller, a tracking index performance (TIP) is defined as follows:

$$TIP = \frac{\sum_{k=1}^{N} |e(k)|}{\sum_{k=1}^{N} |{}^w \hat{v}_b(k)|} \tag{23}$$

Visual controller is based in velocity estimation and tracking error of the object. This index isolates the result of each trial of the particular features of motion of the object (its velocity and shape of motion) that is being tracked. Particularly in this section k means the visual controller sample, $|e(k)|$ is the norm of the tracking error in the k instant, $|{}^w \hat{v}_b(k)|$ is the norm of the estimated velocity of the object in the word coordinate frame.

TABLE 1 Robot kinematic parameters, nominal and calibrated

	Nominal (lengths in mm)			Calibrated (lengths in mm)		
	$i=1$	$i=2$	$i=3$	$i=1$	$i=2$	$i=3$
a_i	500	500	500	500.8946	500.0072	499.9582
b_i	1000	1000	1000	1002.86	1001.37	1001.284
h_i	50	50	50	48.69038	49.95709	48.68006
θ_i	0°	0°	0°	0.3156°	0.8440°	0.9362°
P_x	0			5.310396		
P_y	0			0.316592		
P_z	751.27			743.740498		
ϕ_i	0°	120°	240°	0°	120°	240°
H_i	210	210	210	210	210	210

Parameters that were not estimated.

Both parameters collections, calibrated and not calibrated, are considered into the robot kinematics in order to test the visual controller of the robot. Controller performance is obtained by means of index in eq. (22) (TIP index), results are shown in table 2. Indexes show how the error of the visual controller increases along several trajectories of the object. Error of the visual controller is increased considerably although kinematic errors are small in comparison to the nominal parameters.

TABLE 2 TIP index results and kinematic parameters. Velocity of the object in the tests is less than $1 \, m/s$.

	Nominal	Calibrated
TIP	0.54344	0.43732

CONCLUSIONS

In this paper a calibration method was designed in order to improve accuracy of a parallel robot that is inspired in the Delta robot. The method was tested by means of a visual controller, errors in the controller and the velocity of an object were used to obtain an index. Thus both sets (nominal and corrected) of kinematic parameters were compared. Calibrated parameters have strong influence in the performance of the error in the visual controller. Therefore performance of a tracking visual controller is improved although correction of the parameters is extremely small (errors in the controller are reduced more than 20% when the robot is calibrated). Improvements in the controller performance can be explained as follows: Image visual controller makes use of the robot Jacobian and robot kinematic models (inverse and direct), when the controller corrects the robot trajectory, errors in the model have direct influence in the direction in which the controller tries to correct. On the other hand if the position of the object remains fixed, error in the visual controller converges to cero but the evolution of the error is different for two set of parameters. In this paper three main topics were discussed: 1.-Obtaining of a kinematic model based on incremental and measurements. 2.-Obtaining of a 3D measurement method from one camera in hand. 3.-Obtaining of the influence of small kinematical errors in a classical visual controller.

Due to the influence of kinematic errors in the visual controller, future works are going to concentrate on studying if it is possible to directly calibrate the robot by minimizing the errors in the visual controller. Above approach brings out other problems as can be the addition of the time in the model. Future works include Ping-Pong playing and some videos are shown in the web page:

http://www.disam.upm.es/vision/projects/robotenis/indexI.html

REFERENCES

[1] F. Chaumette and S. Hutchinson, Visual servo control. I. Basic approaches, 2006.

[2] G. Ecorchard and P. Maurine, "Self-calibration of delta parallel robots with elastic deformation compensation," 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005.(IROS 2005), Ieee, 2005, p. 1283–1288.

[3] A. Watanabe, S. Sakakibara, K. Ban, M. Yamada, G. Shen, and T. Arai, "A Kinematic Calibration Method for Industrial Robots Using Autonomous Visual Measurement," *CIRP Annals – Manufacturing Technology*, vol. 55, 2006, pp. 1-6.

[4] A. Nahvi, J. Hollerbach, and V. Hayward, Calibration of a parallel robot using multiple kinematic closed loops, IEEE Comput. Soc. Press,

[5] J. Ouyang, "Ball array calibration on a coordinate measuring machine using a gage block," Measurement, vol. 16, 1995, pp. 219-229.

[6] S. Bai and M.Y. Teo, Kinematic Calibration and Pose Measurement of a Medical Parallel Manipulator by Optical Position Sensors, 2003.

[7] T.Y. Gao, Meng; Li, Calibration method and experiment of Stewart platform using a laser tracker, Ieee, 2003.

[8] E. Amit J., Patel; Kornel F., Calibration of a hexapod machine tool using a redundant leg, 2000.

[9] N. Fazenda, E. Lubrano, S. Rossopoulos, and R. Clavel, "Calibration of the 6 DOF high-precision flexure parallel robot "Sigma 6"," Proceedings of 5th Parallel Kinematics Seminar. Chemnitz, Germany, 2006, p. 379–398.

[10] B. Karlsson and T. Brogårdh, "A new calibration method for industrial robots," Robotica, vol. 19, 2001, p. 691–693.

[11] X. Ren, Z. Feng, and C. Su, "A new calibration method for parallel kinematics machine tools using orientation constraint," International Journal of Machine Tools and Manufacture, vol. 49, 2009, pp. 708-721.

[12] O. Masory, Kinematic calibration of a Stewart platform using pose measurements obtained by a single theodolite, IEEE Comput. Soc. Press, 1995.

[13] K. Großmann, B. Kauschinger, and S. Szatmári, Kinematic calibration of a hexapod of simple design, 2008.

[14] F. Y., Takeda; G., Shen; H., "A DBB-based kinematic calibration method for in-parallel actuated mechanisms using a Fourier series," Transactions- American Society Of Mechanical Engineers Journal Of Mechanical Design, vol. 126, 2004, pp. 856-865.

[15] H. Chanal, E. Duc, P. Ray, and J. Hascoet, "A new approach for the geometrical calibration of parallel kinematics machines tools based on the machining of a dedicated part," International Journal of Machine Tools and Manufacture, vol. 47, 2007, pp. 1151-1163.

[16] G. Canepa, J.M. Hollerbach, and A. Boelen, Kinematic calibration by means of a triaxial accelerometer, IEEE Comput. Soc. Press, 1994.

[17] M. Jose Mauricio S.T., Motta; Guilherme C., De Carvalhob; R.S., Robot calibration using a 3D vision-based measurement system with a single camera, 2001.

[18] Y. Meng and H. Zhuang, "Autonomous robot calibration using vision technology," Robotics and Computer-Integrated Manufacturing, vol. 23, 2007, pp. 436-446.

[19] N. Andreff and P. Martinet, Vision-based self-calibration and control of parallel kinematic mechanisms without proprioceptive sensing, Intel Serv Robotics, 2009.

[20] P. Renaud, N. Andreff, J. Lavest, and M. Dhome, "Simplifying the kinematic calibration of parallel mechanisms using vision-based metrology," IEEE Transactions on robotics, vol. 22, 2006, p. 12–22.

[21] A. M. Y., Amirat; J., Pontnau; F., "A three-dimensional measurement system for robot applications," Journal of Intelligent and Robotic Systems, vol. 9, 1994, pp. 291-299.

[22] Z. Zhang, "A flexible new technique for camera calibration," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, 2002, pp. 1-21.

[23] R. Clavel, "Conception d'un robot parallele rapide a4 degrés de liberté''," These de Doctorat, 1991.

[24] R. Stamper, "degree of freedom parallel manipulator with only translational degrees of freedom," 1997.

[25] J. Sebastian, A. Traslosheros, L. Angel, F. Roberti, and R. Carelli, "Parallel Robot High Speed Object Tracking," Lecture Notes in Computer Science, vol. 4633, 2007, p. 295.

[26] A. Traslosheros, L. Angel, R. Vaca and R.C. J. M. Sebastián, F. Roberti, "New visual Servoing control strategies in tracking tasks using a PKM," Mechatronic Systems, A. Lazinica, Vienna, Austria: In-Tech, .

[27] S. Besnard and W. Khalil, "Identifiable parameters for parallel robots kinematic calibration," IEEE International Conference on Robotics and Automation, IEEE; 1999, 2001, p. 2859–2866.

[28] L. Angel, A. Traslosheros, J. Sebastian, L. Pari, R. Carelli, and F. Roberti, "Vision-Based Control of the RoboTenis System," LECTURE NOTES IN CONTROL AND INFORMATION SCIENCES, vol. 370, 2008, p. 229.