We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS BOOK CITATION INDEX INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

6

Robot Calibration: Modeling Measurement and Applications

José Maurício S. T. Motta
University of Brasilia
Brazil

1. Introduction

Most currently used industrial robots are still programmed by a teach pendant, especially in the automotive industry. However the importance of off-line programming in industry as an alternative to teach-in programming is steadily increasing. The main reason for this trend is the need to minimize machine downtime and thus to improve the rate of robot utilization. Nonetheless, for a successful accomplishment of off-line programming the robots need to be not only repeatable but also accurate.

In addition to improving robot accuracy through software (rather than by changing the mechanical structure or design of the robot), calibration techniques can also minimize the risk of having to change application programs due to slight changes or drifts (wearing of parts, dimension drifts or tolerances, and component replacement effects) in the robot system. This is mostly important in applications that may involve a large number of task points.

Theoretically, with the availability of robot calibration systems integrated with off-line programming systems, it would be possible to implement off-line programming in industry (currently they are not ready to be used in large scale), where multiple robots are used, by programming only one robot and copying the programs from one robot to the others. One of the most evident advantages of this type of programming is the reduction in maintenance expenditure, since the robot can return faster to service and easy reprogramming means the software becomes operational with little effort.

Robot calibration is an integrated process of modeling, measurement, numeric identification of actual physical characteristics of a robot, and implementation of a new model. The calibration procedure first involves the development of a kinematic model whose parameters represent accurately the actual robot. Next, specifically selected robot characteristics are measured using measurement instruments with known accuracy. Then a parameter identification procedure is used to compute the set of parameter values which, when introduced in the robot nominal model, accurately represents the measured robot behavior. Finally, the model in the position control software is corrected.

Many factors such as numerical problems, measurement system deficiencies, cumbersome setups and commercial interests have defied the confidence of the industry in such systems, especially if one realizes that a robot will not normally need full calibration more than once or twice a year.

Source: Industrial Robotics: Programming, Simulation and Applicationl, ISBN 3-86611-286-6, pp. 702, ARS/plV, Germany, December 2006, Edited by: Low Kin Huat

The main objective of this chapter is to present and discuss several theoretical and practical aspects involving methods and systems for robot calibration. Firstly, it is discussed the implementation of techniques to optimize kinematic models for robot calibration through numerical optimization of the mathematical model. The optimized model is then used to compensate the model errors in an off-line programming system, enhancing significantly the robot kinematic model accuracy. The optimized model can be constructed in an easy and straight operation, through automatic assignment of joint coordinate systems and geometric parameter to the robot links. Assignment of coordinate systems by this technique avoids model singularities that usually spoil robot calibration results. Secondly, calibration results are discussed using a Coordinate Measuring Arm as a measurement system on an ABB IRB 2000 Robot.
Further in the chapter it is presented and discussed a 3-D vision-based measurement system developed specifically for robot calibration requirements showing up to be a feasible alternative for high cost measurement systems. The measurement system is portable, accurate and low cost, consisting of a single CCD camera mounted on the robot tool flange to measure the robot end-effector's pose relative to a world coordinate system. Radial lens distortion is included in the photogrammetric model. Scale factors and image centers are obtained with innovative techniques, making use of a multiview approach. Experimentation is performed on three industrial robots to test their position accuracy improvement using the calibration system proposed: an ABB IRB-2400, IRB 6400 and a PUMA-500. The proposed off-line robot calibration system is fast, accurate and ease to setup.
Finally, many theoretical and practical aspects are discussed concerning the relationship between measuring volumes, positions, robot types and measurement systems and the final accuracy expected after the calibration process.

2. Robot Calibration Models for Off-line Programming

Off-line programming is, by definition, the technique of generating a robot program without using a real machine. It presents several advantages over the on-line method. However, there are inevitably differences between the computer model used to perform the graphic simulation and the real world. This is because the effective use of off-line programming in industrial robots requires, additionally, a knowledge of tolerances of the manufacturing components in order to enable realistic planning, i.e. to reduce the gap between simulation and reality. In an actual robot system programming this is still a relatively difficult task and the generated programs still require manual on-line modification at the shop floor. A typical welding line with 30 robots and 40 welding spots per robot takes about 400 hours for robot teaching (Bernhardt, 1997). The difficulties are not only in the determination of how the robot can perform correctly its function, but also for it to be able to achieve accurately a desired location in the workspace. Robot pose errors are attributed to several sources, including the constant (or configuration-independent) errors in parameters (link lengths and joint offsets), deviations which vary predictably with position (e.g., compliance, gear transmission errors) and random errors (e.g., due to the finite resolution of joint encoders). Constant errors are referred to as geometric errors and variable errors are referred to as non-geometric errors (Roth, Mooring and Ravani, 1987). According to Bernhardt (1997) and Schröer (1993), constant errors represent approximately 90% of the overall robot pose errors. Industrial robots usually show pose errors from about 5 to 15mm, even when they are new, and after proper calibration these error can be reduced to about less than 0.5mm (Bernhardt, 1997, Motta, Carvalho and McMaster, 2001).

The Robot Calibration problem has been investigated for more than two decades, but some of its obstacles are still around. Usually, one can tackle the problem implementing model or modeless methods. Modeless methods does not need any kinematic model, using only a grid of known points located in the robot workspace as a standard calibration board. The robot is moved through all the grid points, and the position errors are stored for future compensation by using a bilinear interpolation method and polynomial fitting (Zhuang & Roth, 1996, Park, Xu and Mills, 2002) or error mapping (Bai and Wang, 2004). Although modeless methods are simpler, the calibrated workspace region is small, and each time the robot is to work off that region the calibration process has to be repeated. On the other side, model methods allow a large workspace region to be calibrated, leading to full model calibration. However, important to model methods is an accurate kinematic model that is complete, minimal and continuous and has identifiable parameters (Schröer, Albright and Grethlein, 1997). Researchers have used specific kinematic models that depend on a particular robot geometry and/or calibration method. Model identifiability has already been addressed (e.g., Everett and Hsu, 1988, Zhuang, 1992), and Motta and McMaster (1999) and Motta, Carvalho e McMaster (2001) have shown experimental and simulation results using a rational technique to find an optimal model for a specific joint configuration, requiring a few number of measurement points (for a 6 DOF robot only 15 measurement points) for a model with only geometric parameters (30), in opposition to hundreds of measurement points claimed by other authors (Drouet et al., 2002, Park, Xu and Mills, 2002). A model with singularities or quasi-singular parameterization turns the identification process to be ill-conditioned, leading to solutions that cannot satisfy accuracy requirements when the manipulator is to move off the measurement points. Furthermore, time to convergence increases or may not exist convergence at all, and the number of measurement points may be ten-fold larger than the necessary (Motta, 1999).

3. A Singularity-Free Approach for Kinematic Models

Single minimal modeling convention that can be applied uniformly to all possible robot geometries cannot exist owing to fundamental topological reasons concerning mappings from Euclidean vectors to spheres (Schröer, 1993). However, after investigating many topological problems in robots, concerning inverse kinematics and singularities, Baker (1990) suggested that the availability of an assortment of methods for determining whether or not inverse kinematic functions can be defined on various subsets of the operational spaces would be useful, but even more important, a collection of methods by which inverse functions can actually be constructed in specific situations. Another insightful paper about robot topologies was published by Gottlieb (1986), who noted that inverse functions can never be entirely successful in circumventing the problems of singularities when pointing or orienting.

Mathematically, model-continuity is equivalent to continuity of the inverse function T^{-1}, where T is the product of elementary transformations (rotation and translation) between joints. From this, the definition of parameterization's singularity can be stated as a transformation $T_s \in E$ (parameterization's space of the Euclidean Group - 3 rotations and 3 translations), where the parameter vector $p \in R^6$ (p represents distance or angle) exists such that the rank of the Jacobian $J_s = dT_s/dp$ is smaller than 6. In other way, each parameterization T can be investigated concerning their singularities detecting the zeroes of determinant $\det(J^T.J)$ considered as a function of parameter p.

Thus, investigating the main kinematic modeling conventions one can represent the transformation between links in the Euclidean Group as

$$T = T_x(p_x).T_y(p_y).T_z(p_z).R_z(\gamma).R_y(\beta).R_x(\alpha) \quad (1)$$

where p_x, p_y, p_z are translation coordinates and α, β, γ are rotation coordinates for the axes x, y and z respectively. Then,

$$T = \begin{bmatrix} C\gamma.C\beta & C\gamma.S\beta.S\alpha - S\gamma.C\alpha & C\gamma.S\beta.C\alpha + S\gamma.S\alpha & Px \\ S\gamma.C\beta & S\gamma.S\beta.S\alpha + C\gamma.C\alpha & S\gamma.S\beta.C\alpha - C\gamma.S\alpha & Py \\ -S\beta & C\beta.S\alpha & C\beta.C\alpha & Pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

where C=cos( ) and S=sin( ).
In a more simple symbolic form it can be represented as

$$T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

which after the decomposition of the singularities through the Jacobian determinant results in a manifold of singularities

$$T_s = \begin{bmatrix} 0 & o_x & a_x & p_x \\ 0 & o_y & a_y & p_y \\ \pm 1 & 0 & 0 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

This manifold represents $\gamma = \pm\pi/2$ and $\beta = \pm\pi/2$ (eq. 2), which means there is a singularity when y and z axes are parallel.
The Denavit-Hartemberg (D-H) convention (parameterization) (Paul, 1981) leads to an elementary transformation represented by

$$T(\theta,p_z,p_x,\alpha) = R_z(\theta).T_z(p_z).T_x(p_x).R_x(\alpha) \quad (5)$$

Following the same procedure as before the manifold of singularities is

$$T_s = \begin{bmatrix} n_x & o_x & 0 & p_x \\ n_y & o_y & 0 & p_y \\ n_z & o_z & \pm 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

This result consists of all elements represented as parallel rotary joints. This can be verified by observing the third column showing the representation of one joint axis (z) into the previous one.
The Hayati-convention (Hayati & Mirmirani, 1985) can be represented by

$$T(\theta,p_x,\alpha,\beta) = R_z(\theta).T_x(p_x).R_x(\alpha).R_y(\beta) \quad (7)$$

There are two manifolds of singularities,

$$T_s = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ or } T_s = \begin{bmatrix} n_x & o_x & \lambda p_x & p_x \\ n_y & o_y & \lambda p_y & p_y \\ n_z & o_z & \lambda p_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

These representations show that if the distal *z* axis is in the same *x*-*y* plane of the proximal coordinate system (so perpendicular) or points to its origin, then there is a singularity.
The Veitschegger convention (Veitschegger & Wu, 1986) is a 5-dimensional parameterization as

$$T = R_z(\theta).T_z(p_z).T_x(p_x).R_x(\alpha).R_y(\beta) \tag{9}$$

The manifolds of singularities are

$$T_s = \begin{bmatrix} n_x & o_x & a_x & 0 \\ n_y & o_y & a_y & 0 \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ or } T_s = \begin{bmatrix} n_x & o_x & a_x & \lambda a_x \\ n_y & o_y & a_y & \lambda a_y \\ n_z & o_z & 0 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

These representations show that if the origin of the distal joint coordinate system is on the *z*-axis of the proximal one, or the *z* axes are perpendicular and intercept each other, then there is a singularity. However this convention is not minimal.
Using the same technique presented so far for prismatic joints sets of parameterizations can be used so fourth. The results can be outlined in a list together with their application ranges. The set is a complete, minimal and model-continuous kinematic model (Schröer et al., 1997). The transformations are modeled from a current frame C to a revolute joint, J_R, or to a prismatic joint, J_P, or a fixed and arbitrarily located target frame, TCP-frame. Some of them are not unique since other solutions can fulfill the requirements. Additional elementary transformations are possible, but any added elementary transformation is redundant and thus, cannot be identified. A non-expanded model can always be found, which describes the same kinematic structure as the expanded model.
Elementary transformations that are part of the complete, minimal and model-continuous sub-model being identified are marked bold, i.e. those elementary transformations including model parameters to be identified (errors): (symbols ⊥ and || mean orthogonal and parallel axes respectively)

- Transformation from robot base frame (B) to first joint where joint is translational (J_T):

$$B \perp J_T: \; P = (T_X, T_Y, T_Z, \mathbf{R_Z}, \mathbf{R_X}) \tag{11}$$
$$B \,||\, J_T: \; P = (T_X, T_Y, T_Z, \mathbf{R_X}, \mathbf{R_Y}) \tag{12}$$

And, if first joint is rotational (J_R):

$$B \perp J_R: \; P_X = (\mathbf{T_Y}, \mathbf{T_Z}, \mathbf{R_Z}, \mathbf{R_X}) \tag{13}$$
$$P_Y = (\mathbf{T_X}, \mathbf{T_Z}, \mathbf{R_Z}, \mathbf{R_X}) \tag{14}$$

(If joint axis is near **x**-axis of frame B)

$$B \,||\, J_R: \; P_Z = (\mathbf{T_X}, \mathbf{T_Y}, \mathbf{R_X}, \mathbf{R_Y}) \tag{15}$$

(If joint axis is near **y**-axis of frame B)

- Transformations between consecutive joint frames:

$$J_R \perp J_R: \; P = (\mathbf{R_Z}, \mathbf{T_Z}, \mathbf{T_X}, \mathbf{R_X}, T_Z) \tag{16}$$

(D-H parameterization)

$$J_R \,||\, J_R: \; P = (\mathbf{R_Z}, \mathbf{T_X}, \mathbf{R_X}, \mathbf{R_Y}, T_Z) \tag{17}$$

(assumption: joint axes are not identical)
(Hayati parameterization)

$$J_T \perp J_R: \quad P = (\mathbf{T_Z, R_Z, T_X, R_X,} T_Z) \tag{18}$$

$$J_T \,||\, J_R: \quad P = (T_Z, \mathbf{T_X, T_Y, R_X, R_Y,} T_Z) \tag{19}$$

$$J_T \perp J_T: \quad P = (T_Z, T_X, T_Y, \mathbf{R_Z, R_X}) \tag{20}$$

$$J_T \,||\, J_T: \quad P = (T_Z, T_X, T_Y, \mathbf{R_X, R_Y}) \tag{21}$$

$$J_R \perp J_T: \quad P = (\mathbf{R_Z,} T_X, T_Y, T_Z, \mathbf{R_X}) \tag{22}$$

$$J_R \,||\, J_T: \quad P = (R_Z, T_X, T_Y, T_Z, \mathbf{R_X, R_Y}) \tag{23}$$

- Transformation from last joint to TCP (Tool Center Point) :

$$J_T \perp \text{TCP}: \quad P = (\mathbf{T_Z, T_Y, T_X, [R_Z, R_Y, R_Z]}) \tag{24}$$

$$J_T \,||\, \text{TCP}: \quad P = (\mathbf{T_Z, T_Y, T_X, [R_Z, R_Y, R_X]}) \tag{25}$$

$$J_R \perp \text{TCP}: \quad P = (R_Z, \mathbf{T_X, T_Y, T_Z, [R_Z, R_Y, R_Z]}) \tag{26}$$

$$J_R \,||\, \text{TCP}: \quad P = (R_Z, \mathbf{T_X, T_Y, T_Z, [R_Z, R_Y, R_X]}) \tag{27}$$

Parameters in brackets are not identifiable without TCP-orientation measurements. As an example of the application of the equations above a case using eqs. (16) and (17) is shown below, namely, the known Denavit-Hartemberg (D-H) and Hayati parameterizations. The equations are referred to Fig. 1. For the D-H case one can define four cases:
1 – $Z_R(2)$ is parallel in the same direction as X_r:

$$P = R_Z(90^o).T_Z(p_z).T_X(p_y).R_X(90^o).T_Z(p_x) \tag{28}$$

2 - Z_R (3) is parallel in the opposite direction of X_r:

$$P = R_Z(90°).T_Z(p_z).T_X(p_y).R_X(-90°).T_Z(-p_x) \tag{29}$$

3 - Z_R (4) is parallel in the same direction as Y_r:

$$P = R_Z(0°).T_Z(p_z).T_X(p_y).R_X(-90°).T_Z(-p_y) \tag{30}$$

4 - Z_R (5) is parallel in the opposite direction of Y_r:

$$P = R_Z(0°).T_Z(p_z).T_X(p_x).R_X(90°).T_Z(-p_y) \tag{31}$$

For the case of the Hayati parameterization (eq. 17) (valid only for $T_X \neq 0$) if a joint subsequent position needs two translation parameters to be located, (i. e. in the X and Y direction) one of the two has to be vanished, locating the joint frame in a position such that only one remains. Four cases may be assigned:
Supposing only p_x:
1 – $Z_R(0)$ is in the same direction as Z_r:

$$P = R_Z(0^o).T_X(p_x).R_X(0^o).R_Y(0^o).T_Z(p_z) \tag{32}$$

2 – $Z_R(1)$ is in the opposite direction of Z_r:

$$P = R_Z(0^o).T_X(p_x).\ R_X(180^o).R_Y(0^o).T_Z(-p_z) \tag{33}$$

Supposing only p_y:

3 – $Z_R(0)$ is in the same direction as Z_r:

$$P = R_Z(90^o).T_X(p_y).\ R_X(0^o).R_Y(0^o).T_Z(p_z) \tag{34}$$

4 – $Z_R(0)$ is in the opposite direction of Z_r:

$$P = R_Z(90^o).T_X(p_y).\ R_X(0^o).R_Y(0^o).T_Z(-p_z) \tag{35}$$

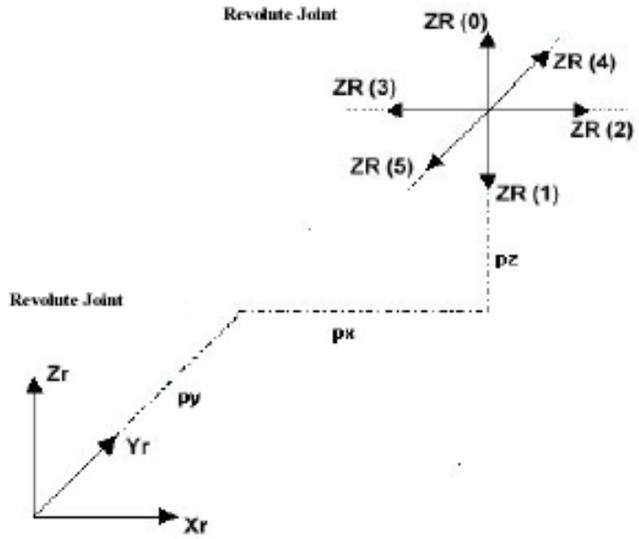


Fig. 1. Frames showing the D-H and Hayati parametrizations.

4. Kinematic Modeling - Assignment of Coordinate Frames

The first step to kinematic modeling is the proper assignment of coordinate frames to each link. Each coordinate system here is orthogonal, and the axes obey the right-hand rule.
For the assignment of coordinate frames to each link one may move the manipulator to its zero position. The zero position of the manipulator is the position where all joint variables are zero. This procedure may be useful to check if the zero positions of the model constructed are the same as those used by the controller, avoiding the need of introducing constant deviations to the joint variables (joint positions).
Subsequently the z-axis of each joint should be made coincident with the joint axis. This convention is used by many authors and in many robot controllers (McKerrow, 1995, Paul, 1981). For a prismatic joint, the direction of the z-axis is in the direction of motion, and its sense is away from the joint. For a revolute joint, the sense of the z-axis is towards the positive direction of rotation around the z-axis. The positive direction of rotation of each joint can be easily found by moving the robot and reading the joint positions on the robot controller display.
According to McKerrow (1995) and Paul (1981), the base coordinate frame (robot reference) may be assigned with axes parallel to the world coordinate frame. The origin of the base frame is coincident with the origin of joint 1 (first joint). This assumes that the axis of the

first joint is normal to the x-y plane. This location for the base frame coincides with many manufacturers' defined base frame.
Afterwards coordinate frames are attached to the link at its distal joint (joint farthest from the base). A frame is internal to the link it is attached to (there is no movements relative to it), and the succeeding link moves relative to it. Thus, coordinate frame *i* is at joint *i+1*, that is, the joint that connects link *i* to link *i+1*.
The origin of the frame is placed as following: if the joint axes of a link intersect, then the origin of the frame attached to the link is placed at the joint axes intersection; if the joint axes are parallel or do not intersect, then the frame origin is placed at the distal joint; subsequently, if a frame origin is described relative to another coordinate frame by using more than one direction, then it must be moved to make use of only one direction if possible. Thus, the frame origins will be described using the minimum number of link parameters.

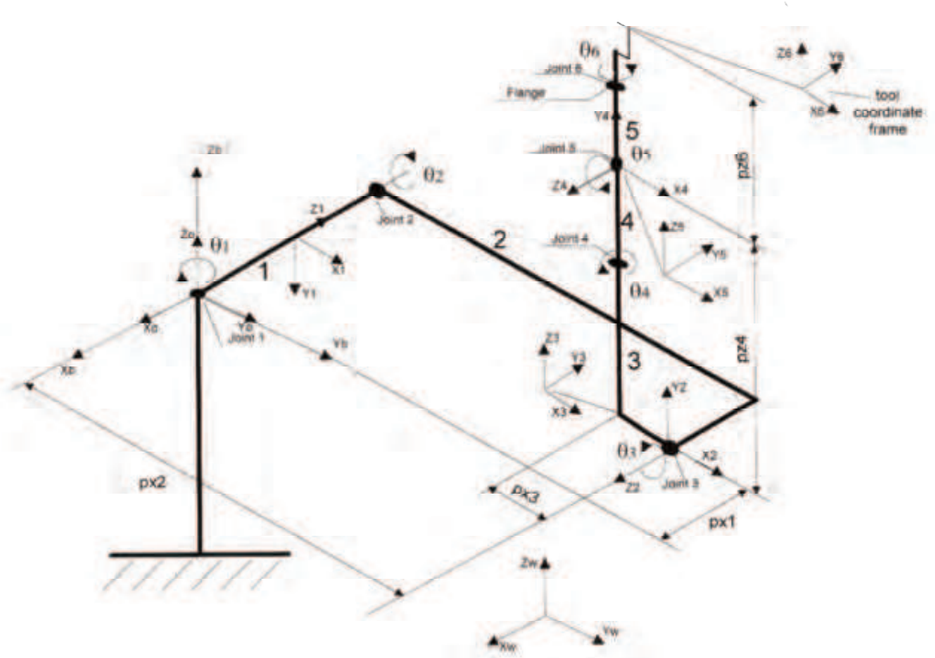


Fig. 2. Skeleton of the PUMA 560 Robot with coordinate frames in the zero position and geometric variables for kinematic modeling. (Out of scale).

The x-axis or the y-axis have their direction according to the convention used to parameterize the transformations between links (e.g. eqs. 16 to 23). At this point the homogeneous transformations between joints must have already been determined. The other axis (x or y) can be determined using the right-hand rule.
A coordinate frame can be attached to the end of the final link, within the end-effector or tool, or it may be necessary to locate this coordinate frame at the tool plate and have a separate hand transformation. The z-axis of the frame is in the same direction as the z-axis of the frame assigned to the last joint (n-1).
The end-effector or tool frame location and orientation is defined according to the controller conventions. Geometric parameters of length are defined to have an index of joint and

direction. The length *pni* is the distance between coordinate frames *i - 1* and *i* , and *n* is the parallel axis in the coordinate system *i - 1*. Figs. 2 and 3 shows the above rules applied to a PUMA-560 and an ABB IRB-2400 robots with all the coordinate frames and geometric features, respectively.

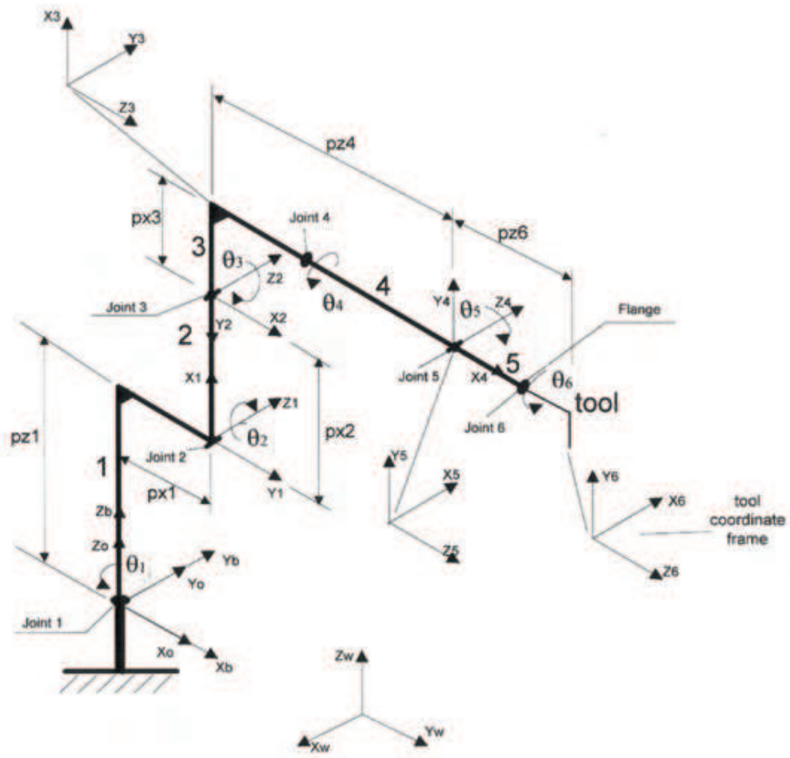


Fig. 3. Skeleton of the ABB IRB-2400 Robot with coordinate frames in the zero position and geometric variables for kinematic modeling. (Out of scale).

5. Kinematic Modeling – Parameter Identification

The kinematic equation of the robot manipulator is obtained by consecutive homogeneous transformations from the base frame to the last frame. Thus,

$$\hat{T}^{0}{}_{N} = \hat{T}^{0}{}_{N}(p) = T^{0}{}_{1}.T^{1}{}_{2}...T^{N-1}{}_{N} = \prod_{i=1}^{N} T^{i-1}{}_{i} \tag{35}$$

where N is the number of joints (or coordinate frames), $p = [p_1^T\ p_2^T\ ...\ p_N^T]^T$ is the parameter vector for the manipulator, and p_i is the link parameter vector for the joint i, including the joint errors. The exact link transformation $A^{i-1}{}_i$ is (Driels & Pathre, 1990):

$$A^{i-1}{}_i = T^{i-1}{}_i + \Delta T_i \quad , \quad \Delta T_i = \Delta T_i(\Delta p_i) \tag{36}$$

where Δp_i is the link parameter error vector for the joint i.
The exact manipulator transformation $\hat{A}^0{}_{N-1}$ is

$$\hat{A}^0{}_N = \prod_{i=1}^{N} (T^{i-1}{}_i + \Delta T_i) = \prod_{i=1}^{N} A^{i-1}{}_i \tag{37}$$

Thus,

$$\hat{A}^0{}_N = \hat{T}^0{}_N + \Delta\hat{T} \quad , \quad \Delta\hat{T} = \Delta\hat{T}(q, \Delta p) \tag{38}$$

where $\Delta p = [\Delta p_1{}^T\ \Delta p_2{}^T \ldots \Delta p_N{}^T]^T$ is the manipulator parameter error vector and $q=[\theta_1{}^T, \theta_2{}^T \theta_N{}^T]^T$ is the vector of joint variables. It must be stated here that $\Delta\hat{T}$ is a non-linear function of the manipulator parameter error vector Δp.
Considering *m* the number of measure positions it can be stated that

$$\hat{A} = \hat{A}^0{}_N = \hat{A}(q, p) \tag{39}$$

where $\hat{A}: \Re^n \times \Re^N$ is function of two vectors with *n* and N dimensions, *n* is the number of parameters and N is the number of joints (including the tool). It follows that

$$\hat{\mathbf{A}} = \hat{\mathbf{A}}^0{}_N = \hat{\mathbf{A}}(\mathbf{q},p) = (\hat{A}(q_1,p),\ldots, \hat{A}(q_m,p))^T: \Re^n \times \Re^{mN} \tag{40}$$

and

$$\Delta\hat{\mathbf{T}} = \Delta\hat{\mathbf{T}}(\mathbf{q}, \Delta p) = (\Delta\hat{T}(q_1, \Delta p),\ldots, \Delta\hat{T}(q_m, \Delta p)^T \quad : \Re^n \times \Re^{mN} \tag{41}$$

All matrices or vectors in bold are functions of *m*. The identification itself is the computation of those model parameter values $p^*=p+\Delta p$ which result in an optimal fit between the actual measured positions and those computed by the model, i.e., the solution of the non-linear equation system

$$\mathbf{B}(\mathbf{q},p^*) = \mathbf{M}(\mathbf{q}) \tag{42}$$

where **B** is a vector formed with position and orientation components of $\hat{\mathbf{A}}$ and

$$\mathbf{M}(\mathbf{q}) = (M(q_1),\ldots, M(q_m))^T \quad \in \Re^{\phi m} \tag{43}$$

are all measured components and ϕ is the number of measurement equations provided by each measured pose. If orientation measurement can be provided by the measurement system then 6 measurement equations can be formulated per each pose. If the measurement system can only measure position, each pose measurement can supply data for 3 measurement equations per pose and then B includes only the position components of $\hat{A}$.
When one is attempting to fit data to a non-linear model, the non-linear least-squares method arises most commonly, particularly in the case that *m* is much larger than *n* (Dennis & Schnabel, 1983). In this case we have from eq. (36), eq. (38) and eq. (42):

$$\mathbf{B}(\mathbf{q}, p^*) = \mathbf{M}(\mathbf{q}) = \mathbf{B}(\mathbf{q}, p) + \mathbf{C}(\mathbf{q}, \Delta p) \tag{44}$$

where **C** is the differential motion vector formed by the position and rotation components of $\Delta\hat{\mathbf{T}}$. From the definition of the Jacobian matrix and ignoring second-order products

$$\mathbf{C}(\mathbf{q}, \Delta p) = \mathbf{J}.\Delta p \tag{45}$$

and so,

$$\mathbf{M}(\mathbf{q}) - \mathbf{B}(\mathbf{q},p) = \mathbf{J}.\Delta p \tag{46}$$

The following notation can be used

$$\mathbf{b} = \mathbf{M}(\mathbf{q}) - \mathbf{B}(\mathbf{q},p) \in \Re^{\phi m} \quad (47)$$

$$\mathbf{J} = \mathbf{J}(\mathbf{q}, \Delta p) \in \Re^{\phi m \times n} \quad (48)$$

$$x = \Delta p \in \Re^{n} \quad (49)$$

$$\mathbf{r} = \mathbf{J}.x - \mathbf{b} \in \Re^{\phi m} \quad (50)$$

Eq. (10) can be solved by a non-linear least-square method in the form

$$\mathbf{J}.x = \mathbf{b} \quad (51)$$

One method to solve non-linear least-square problems proved to be very successful in practice and then recommended for general solutions is the algorithm proposed by Levenberg-Marquardt (LM algorithm) (Dennis & Schnabel, 1983). Several algorithms versions of the L.M. algorithm have been proved to be successful (globally convergent). From eq. (51) the method can be formulated as

$$x_{j+1} = x_j - \left[\mathbf{J}(x_j)^T.\mathbf{J}(x_j) + m_j.\mathbf{I}\right]^{-1}.\mathbf{J}^T(x_j).\mathbf{b}(x_j) \quad (52)$$

where, according to Marquardt suggestion, $\mu_j = 0.001$ if x_j is the initial guess, $\mu_j = \lambda(0.001)$ if $||b(x_{j+1})|| \geq ||b(x_j)||$, $\mu_j = 0.001/\lambda$ if $||b(x_{j+1})|| \leq ||b(x_j)||$ and λ is a constant valid in the range of $2.5 < \lambda < 10$ (Press et al., 1994).

6. Experimental Evaluation

To check the complete system to calibrate robots an experimental evaluation was carried out on an ABB IRB-2000 Robot. This robot was manufactured in 1993 and is used only in laboratory research, with little wearing of mechanical parts due to the low number of hours on work. The robot is very similar to the ABB IRB-2400 Robot, and the differences between both robots exists only in link 1, shown in Fig. 3, where p_{x1} turns to be zero.

6.1. Calibration Volumes and Positions

For this experimental setup different workspace volumes and calibration points were selected, aiming at spanning from large to smaller regions. Five calibration volumes were chosen within the robot workspace, as shown in Fig. 4. The volumes were cubic shaped. In Fig. 5 it is shown the calibration points distributed on the cubic faces of the calibration volumes. The external cubes have 12 calibration points (600mm) and the 3 internal cubes (600, 400 and 200mm) have 27 positions.

The measurement device used was a Coordinate Measuring Arm, (ITG ROMER), with 0,087mm of accuracy, shown in Fig. 6. The experimental routine was ordered in the following sequence: 1) robot positioning; 2) robot joint positions recorded from the robot controller (an interface between the robot controller and an external computer has to be available) and 3) robot positions recorded with the external measuring system. In this experiment only TCP positions were measured, since orientation measuring is not possible with the type of measuring device used. Only few measuring systems have this capacity and some of them are usually based on vision or optical devices. The price of the measuring system appears to be a very important issue for medium size or small companies.

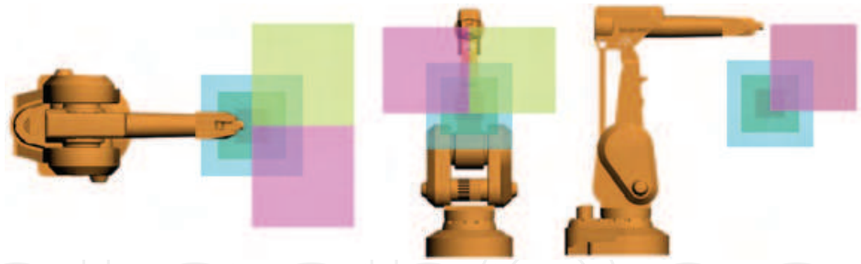


Fig. 4. Workspace Regions where the robot was calibrated.

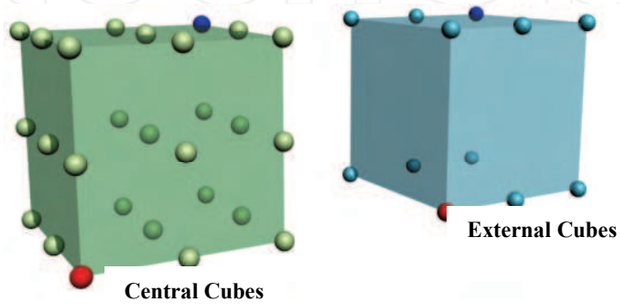


Fig. 5. Cubic calibration volumes and robot positions.

Fig. 6. Coordinate Measuring Arm - ITG ROMER and ABB IRB-2000 manipulator (University of Brasilia).

Fig. 7. represents graphically the calibration results within the regions and volumes of the workspace shown in Fig. 4 with the IRB-2000 Robot. The results presented show that the average of the position errors before and after calibration were higher when the Volumes were larger for both Regions tested. This robot was also calibrated locally, that means the robot was recalibrated in each Region.

A point that deserves attention is that if a robot is calibrated in a sufficiently large calibration volume, the position accuracy can be substantially improved compared to calibration with smaller joint motions. The expected accuracy of a robot in a certain task after calibration is analogous to the evaluation accuracy reported here in various conditions.

Every time a robot moves from a portion of the workspace to another, the base has to be recalibrated. However, in an off-line programmed robot, with or without calibration, that has to be done anyway. If the tool has to be replaced, or after an accident damaging it, it is not necessary to recalibrate the entire robot, only the tool. For that, all that has to be done is to place the tool at few physical marks with known world coordinates (if only the tool is to be calibrated not more than six) and run the off-line calibration system to find the actual tool coordinates represented in the robot base frame.

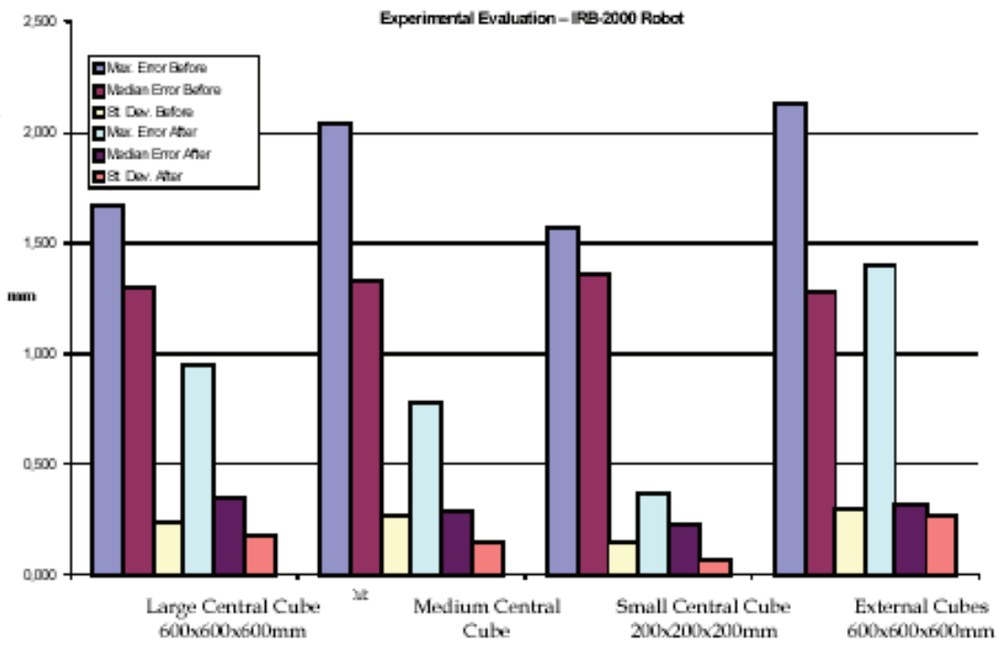


Fig. 7 Experimental evaluation of the robot model accuracy for positioning in each of the volumes.

8. A Vision-Based Measurement System

The main advantages of using a vision-based measurement system for robot calibration are: orientation measurements are feasible, measurement data can be easily recorded for further processing, good potential for high precision, measurements can be adjusted to the scale of the problem and it is a low cost system compared with the very expensive systems based on laser interferometry, theodolites and coordinate measuring arms.

A vision-based measurement system is described here, using a low cost CCD camera and a calibration board of points. The objective of this text is to describe the mathematical model and the experimental assessment of the vision system overall accuracy. The results show very good results for the application and a good potential to be highly improved with a CCD camera with more resolution and with a larger calibration board.

There are basically two typical setups for vision-based robot calibration. The first is to fix cameras in the robot surroundings so that the camera can frame a calibration target mounted on the robot end-effector. The other setup is named hand-mounted camera robot calibration. This latter setup can use a single camera or a pair of cameras. A single moving camera presents the advantages of a large field-of-view with a potential large depth-of-field, and a considerable reduced hardware and software complexity of the system. On the other

hand, a single camera setup needs full camera re-calibration at each pose.
The goal of camera calibration is to develop a mathematical model of the transformation between world points and observed image points resulting from the image formation process. The parameters which affect this mapping can be divided into three categories (Prescott & McLean, 1997, Zhuang & Roth, 1996): a) extrinsic (or external) parameters, which describe the relationship between the camera frame and the world frame, including position (3 parameters) and orientation (3 parameters); b) intrinsic (or internal) parameters, which describe the characteristics of the camera, and include the lens focal length, pixel scale factors, and location of the image center; c) distortion parameters, which describe the geometric nonlinearities of the camera. Some authors include distortion parameters in the group of intrinsic parameters (Tsai, 1987, Weng et al., 1992). Distortion parameters can be present in a model or not.
The algorithm developed here to obtain the camera parameters (intrinsic, extrinsic and distortions) is a two-step method based on the Radial Alignment Constraint (RAC) Model (see Tsai, 1987). It involves a closed-form solution for the external parameters and the effective focal length of the camera. Then, a second stage is used to estimate three parameters: the depth component in the translation vector, the effective focal length, and the radial distortion coefficient. The RAC model is recognized as a good compromise between accuracy and simplicity, which means short processing time (Zhuang & Roth, 1996). Some few modifications were introduced here in the original RAC algorithm, and will be explained later.

8.1 RAC-Based Camera Model

In Fig. 8 the world coordinate system is {xw, yw, zw}; the camera coordinate system is {x, y, z}. The origin of the camera coordinate system is centered at *Oc*, and the z-axis coincides with the optical axis. (X, Y) is the image coordinate system at *Oi* (intersection of the optical axis with the front image plane) and is measured in pixels. (u, v) are the analog coordinates of the object point in the image plane (usually measured in meters or millimeters). (X, Y) lies on a plane parallel to the *x* and *y* axes. *f* is the distance between the front image plane and the optical center.

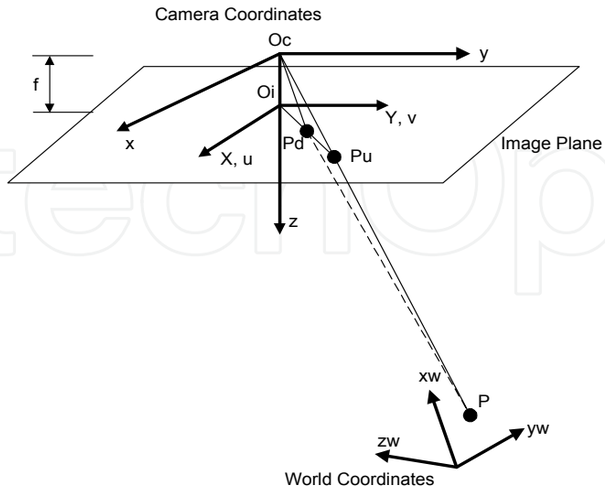


Fig. 8. Pin-hole model and the Radial Alignment Constraint hypothesis.

The rigid body transformation from the object coordinate system (xw, yw, zw) to the camera coordinate system (x, y, z) is

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R.\begin{bmatrix} xw \\ yw \\ zw \end{bmatrix} + T \tag{53}$$

where R is the orthogonal rotation matrix, which aligns the camera coordinate system with the object coordinate system, and it can be represented as

$$R = \begin{bmatrix} r1 & r2 & r3 \\ r4 & r5 & r6 \\ r7 & r8 & r9 \end{bmatrix} \tag{54}$$

and T is the translation vector represented as

$$T = \begin{bmatrix} Tx \\ Ty \\ Tz \end{bmatrix} \tag{55}$$

which is the distance from the origin of the camera coordinate system to the origin of the object coordinate system represented in the camera coordinate system. (x_i, y_i, z_i) is the representation of any point (xw_i, yw_i, zw_i) in the camera coordinate system.

The distortion-free camera model, or "pinhole" model, assumes that every real object point is connected to its correspondent image on the image plane in a straight line passing through the focal point of the camera lens, *Oc*. In Fig. 8 the undistorted image point of the object point, *Pu*, is shown.

The transformation from 3-D coordinate (x, y, z) to the coordinates of the object point in the image plane follows the perspective equations below (Wolf, 1983):

$$u = f.\frac{x}{z} \text{ and } v = f.\frac{y}{z} \tag{56}$$

where *f* is the focal length (in physical units, e.g. millimeters).

The image coordinates (X,Y) are related to (u, v) by the following equations (Zhuang & Roth, 1996, Tsai, 1987):

$$X = s_u.u \text{ and } Y = s_v.v \tag{57}$$

where s_u and s_v are scale factors accounting for TV scanning and timing effects and converting camera coordinates in millimeter or meters to image coordinates (X,Y) in pixels. Lenz and Tsai (1987) relate the hardware timing mismatch between image acquisition hardware and camera scanning hardware, or the imprecision of the timing of TV scanning only to the horizontal scale factor s_u. In eq. (57), s_v can be considered as a conversion factor between different coordinate units.

The scale factors, s_u and s_v, and the focal length, *f*, are considered the intrinsic model parameters of the distortion-free camera model, and reveal the internal information about the camera components and about the interface of the camera to the vision system. The extrinsic parameters are the elements of R and T, which pass on the information about the camera position and orientation with respect to the world coordinate system.

Combining eqs. (56) and (57) follows

$$X = s_u.u = f.s_u.\frac{x}{z} = fx.\frac{x}{z} \tag{58}$$

$$Y = s_v.v = f.s_v.\frac{y}{z} = fy.\frac{y}{z} \tag{59}$$

The equations above combined with eq. (53) produce the distortion-free camera model

$$X = fx.\frac{r1.xw + r2.yw + r3.zw + Tx}{r7.xw + r8.yw + r9.zw + Tz} \tag{60}$$

$$Y = fy.\frac{r4.xw + r5.yw + r6.zw + Ty}{r7.xw + r8.yw + r9.zw + Tz} \tag{61}$$

which relates the world coordinate system (xw, yw, zw) to the image coordinate system (X,Y). *fx* and *fy* are non-dimensional constants defined in eqs. (58) and (59).
Radial distortion can be included in the model as (Tsai, 1987, Weng et al., 1992):

$$X.(1 + k.r^2) = fx.\frac{r1.xw + r2.yw + r3.zw + Tx}{r7.xw + r8.yw + r9.zw + Tz} \tag{62}$$

$$Y.(1 + k.r^2) = fy.\frac{r4.xw + r5.yw + r6.zw + Ty}{r7.xw + r8.yw + r9.zw + Tz} \tag{63}$$

where $r = \mu.X^2 + Y^2$ and μ is the ratio of scale to be defined further in the text. Whenever all distortion effects other than radial lens distortion are zero, a radial alignment constraint (RAC) equation is maintained. In Fig. 8, the distorted image point of the object point, Pd, is shown.
Eqs. (62) and (63) can be linearized as (Zhuang & Roth, 1993):

$$\frac{X}{1 - k.r^2} \cong fx.\frac{r1.xw + r2.yw + r3.zw + Tx}{r7.xw + r8.yw + r9.zw + Tz} \tag{64}$$

$$\frac{Y}{1 - k.r^2} \cong fy.\frac{r1.xw + r2.yw + r3.zw + Ty}{r7.xw + r8.yw + r9.zw + Tz} \tag{65}$$

This transformation can be done under the assumption that $k.r^2 << 1$, i.e. $(1 + k.r^2) \cong (\frac{1}{1 - k.r^2})$ when $k.r^2 << 1$. The first stage to solve eqs. (64) and (65) determines the rotation matrix R (eq. 54), Ty and Tx. The algorithm for that is found in Zhuang & Roth (1996) and Tsai (1987): The second stage is proposed here as

$$\begin{bmatrix} -X_i & x_i & -x_i.r_i^2 \\ -Y_i & \mu.y_i & -\mu.y_i.r_i^2 \end{bmatrix}.\begin{bmatrix} Tz \\ fx \\ k.fx \end{bmatrix} = \begin{bmatrix} X_i.w_i \\ Y_i.w_i \end{bmatrix} \tag{66}$$

where $x_i = r1.xw_i + r2.yw_i + Tx$, $y_i = r4.xw_i + r5.yw_i + Ty$, $w_i = r7.xw_i + r8.yw_i$, and zw_i is made null (all calibration points are coplanar). The overdetermined linear system in eq. (66) can be solved by a linear least-square routine. The calibration points used to solve the system above were on the border of a square grid of 9x9 circular points with 1mm diameter, totting up 32 points. The grid has approximately 200x200mm. The angle between the calibration plane and the image plane should not be smaller than 30 degrees to avoid ill conditioned solutions.
The second stage of the algorithm originally proposed by Tsai (1987) was not a linear solution as it was based on eq.(63), with components only in the *y* direction. Tsai's arguments were based on the fact that *y* components were not contaminated by the mismatch between the frame grabber and camera frequencies. However, it was

experimentally observed during this research that because of the various angles and distances that the images of the target plate were to be acquired from, a linear solution considering both directions (*x* and *y*) showed to be substantially more accurate than using only one direction. The improvement in accuracy of the solutions using eq. (66) was evaluated comparing three different images at different orientations and similar distances, and observing the focus length calculated from the second stage of the RAC algorithm, using eq. (63) (only *y* components), eq. (64) (only *x* components), and eq. (66) (both *x* and *y* components). Ideally, the focus length has to be the same for all images whatever orientation the camera is placed. The use of eq. (66) resulted in considerably closer values of *fx* for the three images than using only one direction (*x* or *y*).

8.2 Calibration of the Scale Factors

A ratio of scale is defined as

$$\mu = \frac{fy}{fx} = \frac{s_v}{s_u} \tag{67}$$

and dividing eq. (64) by (65) yields

$$\frac{X}{Y} = \mu^{-1}.\frac{r1.xw + r2.yw + r3.zw + Tx}{r4.xw + r5.yw + r6.zw + Ty} \tag{68}$$

Two-step camera calibration techniques such as the RAC model can be used to determine the ratio of scale accurately if more than one plane of calibration points is used. This is accomplished by moving a single-plane calibration setup vertically with a z-stage (Zhuang & Roth, 1993). However, if only one coplanar set of calibration points is to be used, pre-calibration of the horizontal scale factor is essential.

Based on the fact that using non-coplanar and parallel planes μ can be determined from two-step calibration methods (Tsai, 1987), several images were obtained from different orientations and positions of the camera. This was accomplished when the camera was mounted on the robot hand during robot calibration measurements. Using the RAC model residuals as calculated from eq. (68) (Lenz & Tsai, 1987, Zhuang et al., 1993):

$$r4.X_i.xw_i + r5.X_i.yw_i + X_i.Ty - r1.Y_i.xw_i - r2.Y_i.yw_i - Y_i.Tx = 0 \tag{69}$$

where *i* is the index representing each calibration point in the image, and based on the fact that □□and the image center are quite independent from each other in the RAC model (a poor guess for one does not affect the determination of the optimal value for the other), the residues found for each image were averaged and then μ was searched to minimize the average of the residuals. The solution found was μ = 0.9825.

8.3 Calibration of the Image Center

The image center is defined as the frame buffer coordinates (Cx, Cy) of the intersection of the optical axis with the image plane. It is usually used as the origin of the imaging process and appears in the perspective equation. In high accuracy applications, it also serves as the center of radially modeled lens distortion (Zhuang & Roth, 1993). The image center is defined from

$$Cx = Xf - X \text{ and } Cy = Yf - Y \tag{70}$$

where (Xf, Yf) is the computed image coordinates for an arbitrary point and (Cx, Cy) is the computed image coordinates for the center Oi in the image plane.

The Radial Alignment Constraint model holds true and is independent of radial lens distortion when the image center is chosen correctly. Otherwise, a residual exists and, unfortunately, the RAC is highly non-linear in terms of the image center coordinates.
The method devised to find the image center was to search for the "best" image center as an average of all images in a sequence of robot measurements, using the RAC residuals. This method could actually find the optimal image center in the model, which could be easily checked calculating the overall robot errors after the calibration. The tests with a coordinate milling machine (explained further in the text) also showed that the determination of the image center by this method led to the best measurement accuracy in each sequence of constant camera orientation.

9 Measurement Accuracy Assessment

The evaluation of the accuracy obtained by the vision measurement system was carried out using a Coordinate Milling Machine (CMM) or any other device that can produce accurate motion on a plane. In this case a CCD camera (Pulnix 6EX – 752x582 pels) was fixed on the CMM's table and moved on pre-defined paths. The calibration board was fixed in front of the camera externally to the CMM, at a position that could allow angles from the camera optical axis to the normal of the calibration plane to be higher than 30^o degrees (avoiding ill-conditioned solutions in the RAC model).
The CMM's table motion produced variations in *z*, *x* and *y* axes of the target plate relative to the camera coordinate system. Fig. 9 shows the coordinate systems of the camera and calibration board.
There were three different measurement sequences of the 25 camera positions. Each sequence was performed with the calibration board placed at different distances and orientations from the camera. The distances were calculated using the photogrammetric model.
The calibration of the remaining intrinsic parameters of the camera, *fx* and *k*, was performed using the first sequence of 25 camera positions, which was chosen to be the closer to the target plate. For each image, values of *fx* and *k* calculated by the algorithm were recorded. Due to noise and geometric inaccuracies each image yielded different values for *fx* and *k*. The average values of the constants *fx* and *k* calculated from the 25 images were 1523 and 7.9×10^{-8} respectively. The standard deviation values for *fx* and *k* were 3.38 and 2.7×10^{-9} respectively.

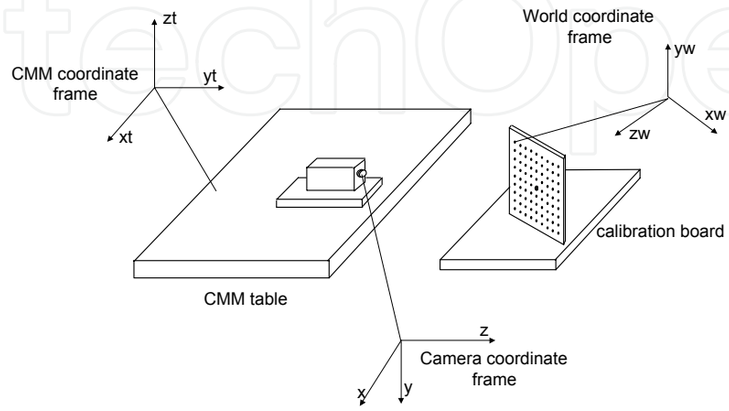


Fig. 9. Diagram showing coordinate systems and the experimental setup for the evaluation of the measurement system accuracy.

The average values for *fx* and *k* should be kept constant from then on, and considered the "best" ones. To check this assumption, the distance traveled by the camera calculated by the Euclidean norm of (XW, YW, ZW) for each camera position, and the distance read from the CMM display were compared to each other. Errors were calculated to be the difference between the distance traveled by the camera using the two methods. It was observed then that the average value of *fx* did not minimize the errors.

Subsequently, an optimal value for *fx* was searched to minimize the errors explained above, by changing slightly the previous one. The value of *k* did not show an important influence on the errors when changed slightly. The optimal values of *fx* and *k* were checked in the same way for the other two measurement sequences at different distances, and showed to produce the best accuracy. The optimal values for *fx* and *k* where found to be 1566 and 7.9 x 10^{-8}, which were kept constant from then on.

Assessment of 3-D measurement accuracy using a single camera is not straightforward. The method designed here to assess the measurement accuracy was to compare each component XW, YW and ZW corresponding to each camera position, to the same vector calculated assuming an average value for the rotation matrix. This method is justified considering that, since there are no rotations between the camera and the calibration board during the tests, the rotation matrix must be ideally the same for each camera position. Since the vector (XW, YW, ZW) depends on R in eq. (54) and T in eq. (55), and as T is calculated from R, all measurement errors would be accounted for in the rotation matrix, apart from the errors due to the ground accuracy of the calibration board. Of course, this assumption is valid only if another method to compare the camera measurements to an external measurement system (in physical units) is used to validate it. That means, the accuracy calculated from the traveled distance validates the accuracy calculated from the rotation matrix. The first method does not consider each error component independently (x, y, z), but serves as a basis to optimize the camera parameters. The second method takes into account the 3-D measurement error, assuming that the "correct" rotation matrix is the average of the ones calculated for each of the 25 positions. The maximum position errors for each measurement sequence were 0.43, 0.97, and 0.72mm respectively. Fig. 10 shows graphically the average, median and standard deviation values of the measurement system position accuracy calculated as explained before, as a function of the average distance from the camera (optical center) to the central point of the calibration board.

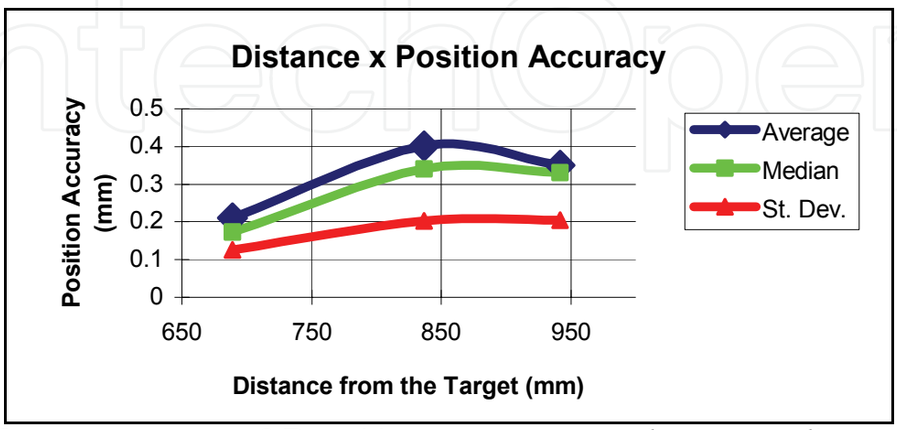


Fig. 10. Measurement system 3-D position accuracy versus the average distance from the camera focal point to the central point of the target.

10 Robot Calibration and Experimental Results

Within the IRB-2400 robot workspace three calibration Regions were defined to collect data, each one with a different Volume (V1, V2 and V3). Fig. 11 represents graphically the three Regions within the robot workspace.

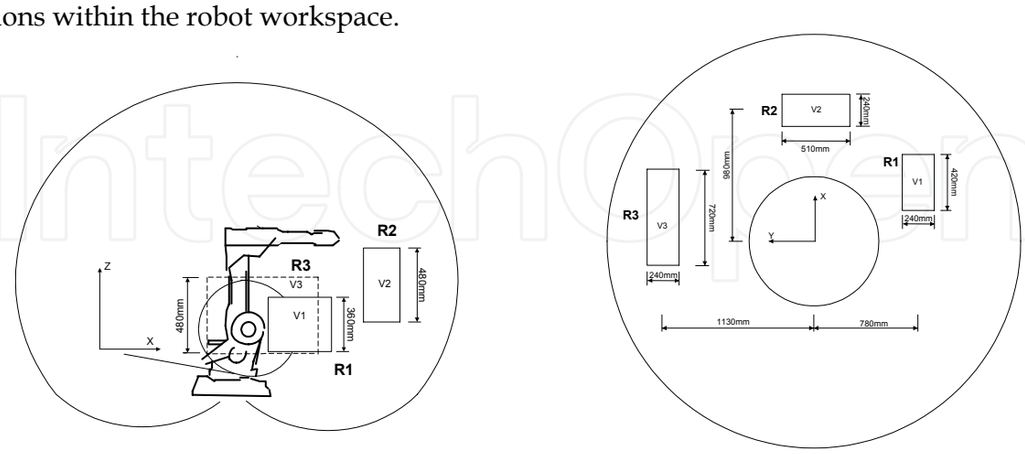


Fig. 11. Side and top view of the IRB-2400 Robot workspace showing Regions, Volumes and their dimensions and locations.

The results from the calculation of average errors and their standard deviation in each Region can be seen in the graphs shown in Fig. 12, calculated before and after the calibration. For calculated and measured data in different coordinate systems to be compared to each other, the robot base coordinate frame was moved to coincide with the world coordinate system at the measurement target plate. This procedure was carried out through a recalibration of the robot base in each Region.
The results presented in Fig. 12 show that the average of the position errors after calibration were very close in value for the three calibration regions tested.

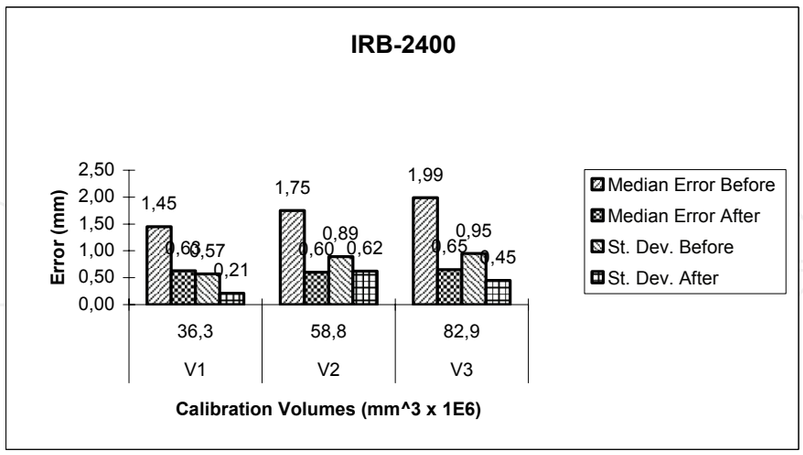


Fig. 12. Average Error and Standard Deviation calculated before and after calibration in each Region.

Within the PUMA-500 robot workspace two calibration Regions were defined to collect data. In each Region three Volumes were defined with different dimensions. Once two different Volumes had the same volume they had also the same dimensions, whatever Region they were in.
Fig. 13 represents graphically all Regions and Volumes within the PUMA-500 workspace.

The results presented in Figs. 15 and 16 show that the average of the position errors before and after calibration were higher when the Volumes were larger for both Regions tested. This robot was also calibrated locally, that means the robot was recalibrated in each Region.

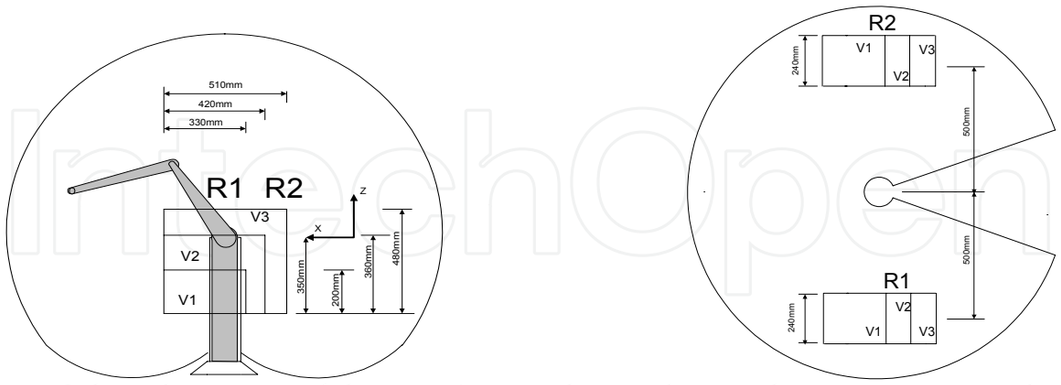


Fig. 13. Side and top view of the PUMA-500 Robot workspace showing Regions, Volumes and their dimensions and locations.

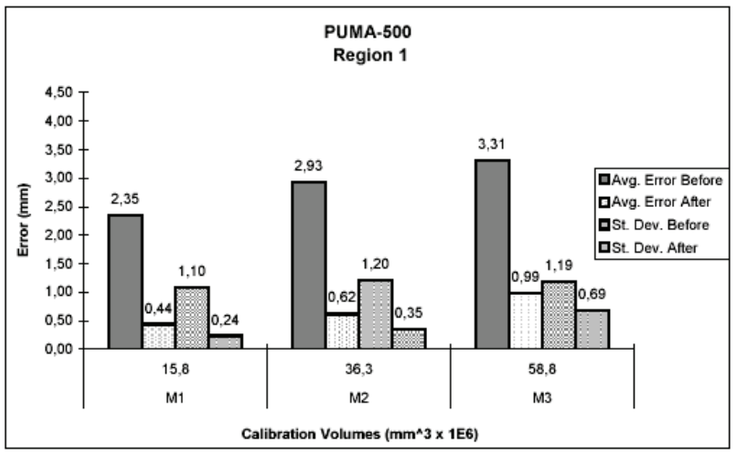


Fig. 14. Average Error and Standard Deviation calculated before and after calibration in each Volume in Region 1.

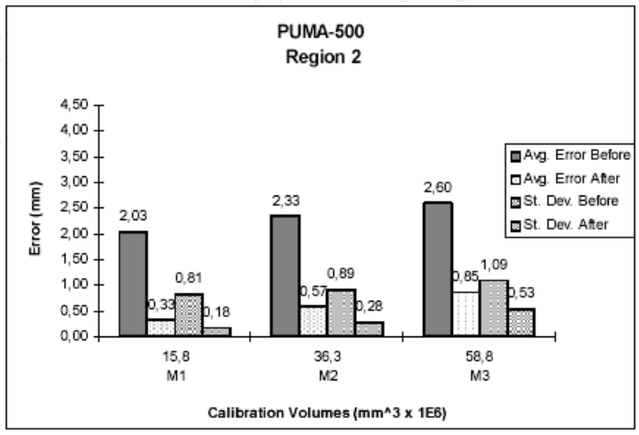


Fig. 15. Average Error and Standard Deviation calculated before and after calibration in each Volume in Region 2.

11. Conclusions and Further Work

The calibration system proposed showed to improve the robot accuracy to well below 1mm. The system allows a large variation in robot configurations, which is essential to proper calibration.
A technique was used and a straightforward convention to build kinematic models for a manipulator was developed, ensuring that no singularities are present in the error model. Mathematical tools were implemented to optimize the kinematic model parameterization, avoiding redundancies between parameters and improving the parameter identification process. A portable, ease of use, speedy and reliable Vision-based measuring system using a single camera and a plane calibration board was developed and tested independently of the robot calibration process.
The robot calibration system approach proposed here stood out to be a feasible alternative to the expensive and complex systems available today in the market, using a single camera and showing good accuracy and ease of use and setup. Results showed that the RAC model used (with slight modifications) is not very robust, since even for images filling the entire screen and captured at approximately the same distances from the target, the focus length was not constant and showed an average value shifted by approximately 3% from the exact one. This amount of error can produce 3-D measurement errors much larger than acceptable. Practically speaking, the solution for this problem developed here for a set of camera and lens was to use an external measurement system to calibrate the camera, at least once. The measurement accuracy obtained is comparable to the best found in academic literature for this type of system, with median values of accuracy of approximately 1:3,000 when compared to the distance from the target. However, this accuracy was obtained at considerable larger distances and different camera orientations than usual applications for cameras require, making the system suitable for robotic metrology.
For future research it is suggested that the target plate and the calibration board have to be improved to permit the camera to be placed at larger ranges of distances from the target, allowing larger calibration volumes to be used. One path that might be followed is to construct a much larger calibration board, with localized clusters of calibration points of different sizes, instead of just one pattern of point distribution. So, if the camera is placed at a greater distance, larger dots can be used all over the area of the calibration board. If the camera is nearer to the target, smaller dots can be used at particular locations on the calibration board. Different dot sizes make easier for the vision processing software to recognize desired clusters of calibration points.
Other sources of lens distortions such as decentering and thin prism can be also modeled, and so their influence on the final measurement accuracy can be understood.
Another issue concerns the influence orientation measured data may have on the final accuracy. Non-geometric parameters such as link elasticity, gear elasticity and gear backlash might be modeled, and a larger number of parameters introduced in the model parameterization. This procedure may improve the accuracy substantially if the robot is used with greater payloads.

12. References

Bai, Y and Wang, D. (2004). Improve the Robot Calibration Accuracy Using a Dynamic Online Fuzzy Error Mapping System, *IEEE Transactions on System, Man, And Cybernetics – Part B: Cybernetics*, Vol. 34, No. 2, pp. 1155-1160.

Baker, D.R. (1990). Some topological problems in robotics, *The Mathematical Intelligencer*,Vol.12, No.1, pp. 66-76.

Bernhardt, R. (1997). Approaches for commissioning time reduction, *Industrial Robot*, Vol. 24, No. 1, pp. 62-71.

Dennis JE, Schnabel RB. (1983). *Numerical Methods for Unconstrained Optimisation and Non-linear Equations*, New Jersey: Prentice-Hall.

Driels MR, Pathre US. (1990). Significance of Observation Strategy on the Design of Robot Calibration Experiments, *Journal of Robotic Systems*, Vol. 7, No. 2, pp. 197-223.

Drouet, Ph., Dubowsky, S., Zeghloul, S. and Mavroidis, C. (2002). Compensation of geometric and elastic errors in large manipulators with an application to a high accuracy medical system, *Robotica*, Vol. 20, pp. 341-352.

Everett, L.J. and Hsu, T-W. (1988). The Theory of Kinematic Parameter Identification for Industrial Robots, *Transaction of ASME*, No. 110, pp. 96-100.

Gottlieb, D.H. (1986). Robots and Topology, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1689-1691.

Hayati, S. and Mirmirani, M., (1985). Improving the Absolute Positioning Accuracy of Robots Manipulators, *Journal of Robotic Systems*, Vol. 2, No. 4, pp. 397-413.

Lenz RK, Tsai RY. (1987). Techniques for Calibration of the Scale Factor and Image Centre for High Accuracy 3D Machine Vision Metrology, *IEEE Proceedings of the International Conference on Robotics and Automation,* pp. 68-75.

McKerrow, P. J. (1995). *Introduction to Robotics*, 1st ed., Ed. Addison Wesley, Singapore.

Motta, J. M. S. T. and McMaster, R. S. (1999). Modeling, Optimizing and Simulating Robot Calibration with Accuracy Imporovements, *Journal of the Brazilian Society of Mechanical Sciences*, Vol. 21, No. 3, pp. 386-402.

Motta, J. M. S. T. (1999). *Optimised Robot Calibration Using a Vision-Based Measurement System with a Single Camera*, Ph.D. thesis, School of Industrial and Manufacturing Science, Cranfield University, UK.

Motta, J. M. S. T., Carvalho, G. C. and McMaster, R. S. (2001), Robot Calibration Using a 3-D Vision-Based Measurement System With a Single Camera, *Robotics and Computer Integrated-Manufacturing*, Ed. Elsevier Science, U.K., Vol. 17, No. 6, pp. 457-467.

Park, E. J., Xu, W and Mills, J. K. (2002). Calibration-based absolute localization of parts for multi-robot assembly, *Robotica*, Vol. 20, pp. 359-366.

Paul, R. P., (1981). *Robot Manipulators - Mathematics, Programming, and Control*, Boston, MIT Press, Massachusetts, USA.

Prescott B, McLean GF. (1997). Line-Based Correction of Radial Lens Distortion, *Graphical Models and Image Processing,* Vol. 59, No. 1, pp. 39-47.

Press WH, Teukolsky SA, Flannery BP, Vetterling WT. (1994). *Numerical Recipes in Pascal - The Art of Scientific Computer*, New York: Cambridge University Press.

Roth, Z.S., Mooring, B.W. and Ravani, B. (1987). An Overview of Robot Calibration, *IEEE Journal of Robotics and Automation*, RA-3, No. 3, pp. 377-85.

Schröer, K. (1993). Theory of kinematic modeling and numerical procedures for robot calibration, *Robot Calibration*, Chapman & Hall, London.

Schröer, K., Albright, S. L. and Grethlein, M. (1997), Complete, Minimal and Model-Continuous Kinematic Models for Robot Calibration, *Robotics & Computer-Integrated Manufacturing*, Vol. 13, No. 1, pp. 73-85.

Tsai RY. (1987). A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the Shelf TV Cameras and Lenses. *IEEE International Journal of Robotics and Automation*, RA-3, No. 4, pp. 323-344.

Veitscheggar, K. W. and Wu, C. W. (1986). Robot Accuracy Analysis based on Kinematics. *IEEE Journal of Robotics and Automation*, Vol. 2, No. 3, pp. 171-179.

Weng J, Cohen P, Herniou M. (1992). Camera Calibration with Distortion Models and Accuracy Evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 10, pp. 965-980.

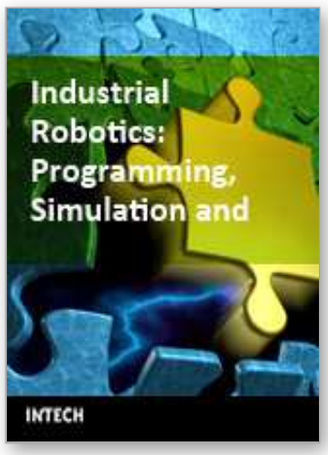
Wolf PR. (1983). *Elements of Photogrammetry*, McGraw-Hill, Singapore.

Zhuang H, Roth ZS, Xu X, Wang K. (1993). Camera Calibration Issues in Robot Calibration with Eye-on-Hand Configuration. *Robotics & Computer Integrated Manufacturing*, Vol. 10, No. 6, pp. 401-412.

Zhuang H, Roth ZS. (1993). A Linear Solution to the Kinematic Parameter Identification of Robot Manipulators. *IEEE Transactions on Robotics and Automation*, Vol. 9, No. 2, pp. 174-185.

Zhuang H, Roth ZS. (1996). *Camera-Aided Robot Calibration*, CRC Press, Boca Raton.Fla, USA.

Zhuang, H. (1992). A Complete and Parametrically Continuous Kinematic Model for Robot Manipulators, *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 4, pp. 451-63.



Industrial Robotics: Programming, Simulation and Applications

Edited by Low Kin Huat

ISBN 3-86611-286-6

Hard cover, 702 pages

Publisher Pro Literatur Verlag, Germany / ARS, Austria

Published online 01, December, 2006

Published in print edition December, 2006

This book covers a wide range of topics relating to advanced industrial robotics, sensors and automation technologies. Although being highly technical and complex in nature, the papers presented in this book represent some of the latest cutting edge technologies and advancements in industrial robotics technology. This book covers topics such as networking, properties of manipulators, forward and inverse robot arm kinematics, motion path-planning, machine vision and many other practical topics too numerous to list here. The authors and editor of this book wish to inspire people, especially young ones, to get involved with robotic and mechatronic engineering technology and to develop new and exciting practical applications, perhaps using the ideas and concepts presented herein.

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen