Robotics and Autonomous Systems

Robotics and Autonomous Systems 18 (1996) 395–410

Autonomous robot calibration using a trigger probe ★

Xiao-Lin Zhong [a,*], John M. Lewis [a], Francis L.N.-Nagy [b]

[a] *Department of Mechanical, Manufacturing and Software Engineering, Napier University, Edinburgh, 10 Colinton Road, Edinburgh, Scotland, UK*

[b] *Information Technology Institute, University of Salford, Salford M5 4WT, UK*

Received 1 February 1995; revised 30 November 1995
Communicated by F.C.A. Groen

Abstract

This paper presents a new robot autonomous calibration method using a trigger probe. The robot grips a simple probe (which was manufactured as a standard end-effector tool) automatically to touch constraint planes in a workspace (the locations of the constraint planes are not necessarily known exactly). The robot internal sensor measurements are recorded for kinematic calibration while the tip-point of the probe is in contact with the constraint plane. The kinematic constraint conditions are obtained from the known shape of the constraint surface, rather than from the measured reference locations in a workspace. The new method eliminates any use of external measuring devices for robot end-effector location measurements for robot calibration; thus it is suitable for a periodic robot re-calibration in a shop–floor environment. Both simulation and experimental results for a six degree-of-freedom (DOF) PUMA robot are given in this paper. The evaluation results using an external precision measuring device – Coordinate Measuring Machine(CMM) – are also presented.

Keywords: Autonomous robot calibration; Trigger probe; Constraint planes

1. Introduction

The positioning accuracy of a robot manipulator is a major concern when using advanced off-line programming methods in a robotic system. The desired locations of a robot end-effector are normally specified in its workspace, while these locations are achieved by controlling its joint values in robot joint space. A kinematic model is built in a robot controller to perform the transformation from workspace to joint space. However, for a number of reasons, including manufacturing tolerance, set-up errors, wear and tear, transmission errors and compliance, etc., the internal kinematic model used in the robot controller will not accurately describe the kinematic transformation of the actual robot. Therefore, the actual locations achieved by controlling joint values obtained from the controller's internal model will deviate from the desired locations. It has been shown that as much as 95% of robot positioning inaccuracy arises from the inaccuracy in its kinematic model description [7]. Robot kinematic calibration is defined as a process of improving robot positioning accuracy through modification of its kinematic control model without changing its hardware configurations [9].

★ Portion of this paper was presented at the *IEEE Int. Conf. on Robotics and Automation (ICRA'95)*, Nagoya, Japan (May 1995).

* Corresponding author.

PII S0921-8890(96)00011-5

There has been an extensive research concerning robot kinematic calibration approaches over the past decades [7,9,14,18,4]. Normally kinematic calibration consists of the following sequential procedures:

(1) Choose a proper kinematic model to describe the relationship between robot joint space and its workspace co-ordinates. Denavit-Hartenburg (D-H) [5] Homogenous transformation is a standard mathematic tool adopting four pair parameters to describe the spatial relationship of manipulator workspace and joint space co-ordinates. It is known that the standard D-H model has a singular problem for robots with two consecutive parallel joints, therefore a modified D-H model using five pair parameters is commonly used in robot calibration [17]. The l th robot end-effector pose vector z_l is represented as a non-linear function of the kinematic parameter vector

$$z_l = f(\boldsymbol{\rho}_l), \tag{1}$$

where $\boldsymbol{\rho}_l = [\boldsymbol{a}, \boldsymbol{d}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\theta}_l]^{\mathrm{T}}$ and $\boldsymbol{a}, \boldsymbol{d}, \boldsymbol{\alpha}$ represent manipulator link length, link offset, and twist angle, respectively; $\boldsymbol{\beta}$ is an introduced rotation parameter which is used to replace the common normal parameter $\boldsymbol{d}$ for consecutive joints with parallel joint axes; and $\boldsymbol{\theta}_l$ is the l th joint variable which is changing from one configuration to another.

(2) Take experimental measurements of robot end-effector locations using external co-ordinate measuring devices. Then the actual measured pose vector z_l^a of the robot end-effector is used to compare with the one predicted by the theoretic model to obtain the workspace inaccuracy vector $\delta \boldsymbol{x}_l$:

$$\delta \boldsymbol{x}_l = z_l^{\mathrm{a}} - f(\boldsymbol{\rho}_l). \tag{2}$$

The collected workspace inaccuracy vectors at different configurations are then aggregated as a single inaccuracy vector $\delta \boldsymbol{x}$ for kinematic identification.

(3) Identify the parameter errors by minimising the workspace inaccuracy vector in the least mean square sense through updating the kinematic parameters in the kinematic model (1). This is an optimisation problem and can be solved by any standard optimisation procedures. By linearising the inaccuracy model (2), the inaccuracy vector is then represented as a linear function of kinematic errors $\Delta \boldsymbol{\rho} = [\Delta \boldsymbol{a}, \Delta \boldsymbol{d}, \Delta \boldsymbol{\alpha}, \Delta \boldsymbol{\beta}, \Delta \boldsymbol{\theta}]^{\mathrm{T}}$:

$$\delta \boldsymbol{x}_l = \boldsymbol{J}_l \, \Delta \boldsymbol{\rho}, \tag{3}$$

where

$$\boldsymbol{J}_l = \left[\frac{\partial f}{\partial \boldsymbol{a}}, \frac{\partial f}{\partial \boldsymbol{d}}, \frac{\partial f}{\partial \boldsymbol{\alpha}}, \frac{\partial f}{\partial \boldsymbol{\beta}} \right] \Bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}_l}$$

is a special Jacobian matrix relating the small changes in kinematic parameters to the pose changes in workspace [12,9].

Aggregating m observations, the kinematic error then can be identified by a linear least square method

$$\Delta \boldsymbol{\rho} = [\boldsymbol{J}^{\mathrm{T}} \, \boldsymbol{J}]^{-1} \boldsymbol{J}^{\mathrm{T}} \delta \boldsymbol{x}, \tag{4}$$

where $\boldsymbol{J} = [\boldsymbol{J}_1, \boldsymbol{J}_2, \ldots, \boldsymbol{J}_m]^{\mathrm{T}}$ is the aggregated special Jacobian matrix, and $\delta \boldsymbol{x} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m]^{\mathrm{T}}$ is the aggregated end-effector inaccuracy vector.

(4) Implement the identified model. Due to the practical difficulties in incorporating the identified model into a robot controller [9], normally the joint inverse compensation scheme is adopted to compensate for the kinematic errors [8,9].

These procedures require extensive human intervention and off-line processing, which precludes the method from being used to perform on-site calibration in an industrial environment at regular intervals. The requirements of measurement phase are particularly demanding, and expensive and complicated pose measuring devices are needed for calibration (such as co-ordinate measuring machines [4,21], theodolites [18], servocontrolled laser interferometers [12], sonic and visual sensors [14,22], etc.). It has been reported that partial pose information (incomplete components of the pose vector) is sufficient for complete parameter identification [15,17,3]. Tang [15] utilised a special fixture to obtain partial pose measurements of a robot end-effector. The fixture consists of a flat plate with some accurately located points on it, and an end-effector with a flat surface and some angle to the last axis of the robot. In the 'free mode' of robot, the robot end-effector was manually moved to the known points of the plate and against the flat plate such that the position of the end-effector and one component of the end-effector orientation were 'measured'. The partial pose measurement scheme eases the requirements for measuring devices, but it still requires intensive human intervention during the set up and measuring process. In addition, as pointed out by Driels [3], not every robot provides a 'free mode' in which the manipulator can be moved manually while the power is on and the joint servosystem is off.

Autonomous robot calibration is defined as the automated process of determining a robot's model by using only its *internal sensors*. It has been observed that autonomous calibrations are possible for robot manipulators with either some a priori knowledge of the task constraint or redundancy of the sensing systems (e.g., two robots gripping together) [1]. The automated data collection schemes were proposed for robot calibration using linear-variable differential transformer (LVDT) ball bar system [6] or wired potentiometer (which can be considered as a flexible ball bar system) [3] connecting the robot end-effector to the known reference point. The closed-loop constraints were formed for kinematic identification by obtaining the accurate radial measurements of the ball bar or the wired potentiometer. But special fixtures are needed for such a system which may require painstaking efforts to set up, and the added fixtures are difficult to model. The autonomous calibration of hand–eye systems has also been performed by using robot joint readings and camera co-ordinate measurements to form the closed-loop constraints [1,22]. The problems for autonomous calibration using camera systems are that not all robotic applications incorporate the visual camera as part of the system; and the camera measurements are known to be insufficiently accurate for manipulator calibration covering a large workspace volume. Another kind of task constraint has been proposed for robot kinematic parameter identification which utilised laser line tracking in the robot workspace [10]. While the motion of robot tip-point was constrained to a line motion in workspace, the robot joint values were recorded for kinematic identification. But only simulation results for a planar two-link manipulator were given in [10].

In this paper, we present a new robot kinematic calibration scheme which can be implemented autonomously and is suitable for on-site calibration in an industrial environment. Gripping a simple trigger probe – *Renishaw probe* (Fig. 2), the robot uses the probe as its extended link to touch constraint planes in its workspace (the locations of the constraint planes are not necessarily known exactly). Only the robot joint readings and the Cartesian position values (reported by controller) are recorded for identification while the tip-point of the probe is touching the constraint plane from various configurations. Neither external sensor measurements nor human interventions are required in the calibration, hence the calibration process is fully autonomous. A linear identification model has been derived from the consistency conditions of a plane, and is presented in Section 2. In Section 3, we describe the data collection method used and the experimental set-up. Both simulation and experimental results for a PUMA 560 robot are given in Section 4. The cross evaluation results using an external global measuring device are also presented. Discussions and conclusions are given in Section 5.

2. Formulation of the Kinematic Identification Model

The objective of kinematic identification is to identify the actual kinematic parameters of the robot manipulator. Let $\Delta\boldsymbol{\rho}$ be the kinematic errors which are assumed to be small perturbations from the nominal values specified by the robot manufacturer, where $\Delta\boldsymbol{\rho}$ is an $n \times 1$ vector, and n is the number of kinematic parameters. Without external measurements, the actual position $\boldsymbol{p}_l = [x_l \ y_l \ z_l]^{\mathrm{T}}$ of the robot end-point remains unknown but must be near the nominal position $\boldsymbol{p}_l^0$ predicted by the robot controller. Using the Taylor series to the first order, we have

$$\begin{aligned}\boldsymbol{p}_l &= \boldsymbol{p}_l^0 + \boldsymbol{J}_l \, \Delta\boldsymbol{\rho} \\ &= [x_l^0 \ \ y_l^0 \ \ z_l^0]^{\mathrm{T}} + [\boldsymbol{j}_l^x \ \boldsymbol{j}_l^y \ \boldsymbol{j}_l^z]^{\mathrm{T}} \, \Delta\boldsymbol{\rho}, \end{aligned} \tag{5}$$

where $\boldsymbol{J}_l$ is a $3 \times n$ matrix which is the positional component of robot special Jacobian, which can be calculated with robot joint readings and nominal kinematic parameters; $\boldsymbol{j}_l^x, \boldsymbol{j}_l^y$ and $\boldsymbol{j}_l^z$ are the x, y and z components of $\boldsymbol{J}_l$, respectively, and l is the subscript index representing different touch points.

Although the exact locations of touch points are unknown, they are constrained to lie on a plane. The consistency condition of a plane leads to the construction of the identification model (Fig. 1). The difference vector between two consecutive touch points is

$$\begin{aligned}\Delta\boldsymbol{p}_l &= \boldsymbol{p}_l - \boldsymbol{p}_{l-1} \\ &= [\Delta x_l^0 \ \ \Delta y_l^0 \ \ \Delta z_l^0]^{\mathrm{T}} + [\boldsymbol{j}_l^x \ \boldsymbol{j}_l^y \ \boldsymbol{j}_l^z]^{\mathrm{T}} \, \Delta\boldsymbol{\rho}, \end{aligned} \tag{6}$$

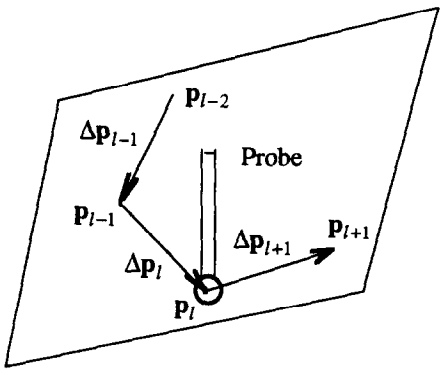


Fig. 1. Constraint conditions for co-planar points.

where

$$\Delta x_l^0 = x_l^0 - x_{l-1}^0, \qquad \Delta y_l^0 = y_l^0 - y_{l-1}^0,$$
$$\Delta z_l^0 = z_l^0 - z_{l-1}^0, \qquad \Delta \boldsymbol{j}_l^x = \boldsymbol{j}_l^x - \boldsymbol{j}_{l-1}^x,$$
$$\Delta \boldsymbol{j}_l^y = \boldsymbol{j}_l^y - \boldsymbol{j}_{l-1}^y, \qquad \Delta \boldsymbol{j}_l^z = \boldsymbol{j}_l^z - \boldsymbol{j}_{l-1}^z.$$

The difference vectors are normalised to roughly unit vectors by dividing the vectors using their nominal length.

$$\Delta \boldsymbol{p}_l = \frac{\boldsymbol{p}_l - \boldsymbol{p}_{l-1}}{\|\boldsymbol{p}_l - \boldsymbol{p}_{l-1}\|} \approx \frac{\boldsymbol{p}_l - \boldsymbol{p}_{l-1}}{\sqrt{(\Delta x_l^0)^2 + (\Delta y_l^0)^2 + (\Delta z_l^0)^2}}. \tag{7}$$

The necessary and sufficient condition for the touch points $\boldsymbol{p}_{l-2}, \boldsymbol{p}_{l-1}, \boldsymbol{p}_l$ and $\boldsymbol{p}_{l+1}$ to lie on one constraint plane is that the volume of the parallelopiped formed by the difference vectors $\Delta\boldsymbol{p}_{l-1}$, $\Delta\boldsymbol{p}_l$ and $\Delta\boldsymbol{p}_{l+1}$ must be equal to zero, i.e., the determinant of the matrix formed by the difference vectors must be zero. Using Eq. (6), we have Eq. (8)

$$\left|\left[\Delta\boldsymbol{p}_{l-1} \;\; \Delta\boldsymbol{p}_l \;\; \Delta\boldsymbol{p}_{l+1}\right]\right| = \begin{vmatrix} \Delta x_{l-1}^0 + \Delta \boldsymbol{j}_{l-1}^x \Delta\rho & \Delta x_l^0 + \Delta \boldsymbol{j}_l^x \Delta\rho & \Delta x_{l+1}^0 + \Delta \boldsymbol{j}_{l+1}^x \Delta\rho \\ \Delta y_{l-1}^0 + \Delta \boldsymbol{j}_{l-1}^y \Delta\rho & \Delta y_l^0 + \Delta \boldsymbol{j}_l^y \Delta\rho & \Delta y_{l+1}^0 + \Delta \boldsymbol{j}_{l+1}^y \Delta\rho \\ \Delta z_{l-1}^0 + \Delta \boldsymbol{j}_{l-1}^z \Delta\rho & \Delta z_l^0 + \Delta \boldsymbol{j}_l^z \Delta\rho & \Delta z_{l+1}^0 + \Delta \boldsymbol{j}_{l+1}^z \Delta\rho \end{vmatrix} = 0. \tag{8}$$

Ignoring the second and higher orders of $\Delta\rho$, Eq. (8) can be written as

$$\begin{vmatrix} \Delta \boldsymbol{j}_{l-1}^x \Delta\rho & \Delta x_l^0 & \Delta x_{l+1}^0 \\ \Delta \boldsymbol{j}_{l-1}^y \Delta\rho & \Delta y_l^0 & \Delta y_{l+1}^0 \\ \Delta \boldsymbol{j}_{l-1}^z \Delta\rho & \Delta z_l^0 & \Delta z_{l+1}^0 \end{vmatrix} + \begin{vmatrix} \Delta x_{l-1}^0 & \Delta \boldsymbol{j}_l^x \Delta\rho & \Delta x_{l+1}^0 \\ \Delta y_{l-1}^0 & \Delta \boldsymbol{j}_l^y \Delta\rho & \Delta y_{l+1}^0 \\ \Delta z_{l-1}^0 & \Delta \boldsymbol{j}_l^z \Delta\rho & \Delta z_{l+1}^0 \end{vmatrix} + \begin{vmatrix} \Delta x_{l-1}^0 & \Delta x_l^0 & \Delta \boldsymbol{j}_{l+1}^x \Delta\rho \\ \Delta y_{l-1}^0 & \Delta y_l^0 & \Delta \boldsymbol{j}_{l+1}^y \Delta\rho \\ \Delta z_{l-1}^0 & \Delta z_l^0 & \Delta \boldsymbol{j}_{l+1}^z \Delta\rho \end{vmatrix} + \begin{vmatrix} \Delta x_{l-1}^0 & \Delta x_l^0 & \Delta x_{l+1}^0 \\ \Delta y_{l-1}^0 & \Delta y_l^0 & \Delta y_{l+1}^0 \\ \Delta z_{l-1}^0 & \Delta z_l^0 & \Delta z_{l+1}^0 \end{vmatrix} = 0. \tag{9}$$

Prior to the expansion of Eq. (9), let us introduce a more compact notation. We define a function $\det(X_1, X_2, X_3)$ which generalises determinant calculation to the situation where one of the three variables X_1, X_2, X_3 is a $3 \times M$ matrix (where M is an integer larger than one), and the other two variables are 3×1 vector. The $\det(X_1, X_2, X_3)$ will return a matrix of $1 \times M$, whose element resulting from the determinant of the matrix formed by the corresponding *column vector* of the matrix variable and the other two vector variables. For example

$$\det\left(\begin{matrix} x_1 \\ y_1 \\ z_1 \end{matrix} \begin{bmatrix} x_{21} & x_{22} \\ y_{21} & y_{22} \\ z_{21} & z_{22} \end{bmatrix} \begin{matrix} x_3 \\ y_3 \\ z_3 \end{matrix}\right) = \begin{bmatrix} \begin{vmatrix} x_1 & x_{21} & x_3 \\ y_1 & y_{21} & y_3 \\ z_1 & y_{21} & z_3 \end{vmatrix} & \begin{vmatrix} x_1 & x_{22} & x_3 \\ y_1 & y_{22} & y_3 \\ z_1 & z_{22} & z_3 \end{vmatrix} \end{bmatrix}. \tag{10}$$

Using the definition of $\det(\bullet)$, Eq. (9) is written as

$$\left(\det\begin{pmatrix} \Delta \boldsymbol{j}_{l-1}^x & \Delta x_l^0 & \Delta x_{l+1}^0 \\ \Delta \boldsymbol{j}_{l-1}^y & \Delta y_l^0 & \Delta y_{l+1}^0 \\ \Delta \boldsymbol{j}_{l-1}^z & \Delta z_l^0 & \Delta z_{l+1}^0 \end{pmatrix} + \det\begin{pmatrix} \Delta x_{l-1}^0 & \Delta \boldsymbol{j}_l^x & \Delta x_{l+1}^0 \\ \Delta y_{l-1}^0 & \Delta \boldsymbol{j}_l^y & \Delta y_{l+1}^0 \\ \Delta z_{l-1}^0 & \Delta \boldsymbol{j}_l^z & \Delta z_{l+1}^0 \end{pmatrix} + \det\begin{pmatrix} \Delta x_{l-1}^0 & \Delta x_l^0 & \Delta \boldsymbol{j}_{l+1}^x \\ \Delta y_{l-1}^0 & \Delta y_l^0 & \Delta \boldsymbol{j}_{l+1}^y \\ \Delta z_{l-1}^0 & \Delta z_l^0 & \Delta \boldsymbol{j}_{l+1}^z \end{pmatrix}\right)\Delta\rho + \begin{vmatrix} \Delta x_{l-1}^0 & \Delta x_l^0 & \Delta x_{l+1}^0 \\ \Delta y_{l-1}^0 & \Delta y_l^0 & \Delta y_{l+1}^0 \\ \Delta z_{l-1}^0 & \Delta z_l^0 & \Delta z_{l+1}^0 \end{vmatrix} = 0. \tag{11}$$

Note that in Eq. (11) $\Delta\boldsymbol{J}_l = [\Delta \boldsymbol{j}_l^x \;\; \Delta \boldsymbol{j}_l^y \;\; \Delta \boldsymbol{j}_l^z]^{\mathrm{T}}$ is a $3 \times n$ matrix, as well as $\Delta\boldsymbol{J}_{l-1}$ and $\Delta\boldsymbol{J}_{l+1}$. Denoting

$$\Delta X_l = \begin{vmatrix} \Delta x_{l-1}^0 & \Delta x_l^0 & \Delta x_{l+1}^0 \\ \Delta y_{l-1}^0 & \Delta y_l^0 & \Delta y_{l+1}^0 \\ \Delta z_{l-1}^0 & \Delta z_l^0 & \Delta z_{l+1}^0 \end{vmatrix},$$

$$\boldsymbol{H}_l = \det\begin{pmatrix} \Delta \boldsymbol{j}_{l-1}^x & \Delta x_l^0 & \Delta x_{l+1}^0 \\ \Delta \boldsymbol{j}_{l-1}^y & \Delta y_l^0 & \Delta y_{l+1}^0 \\ \Delta \boldsymbol{j}_{l-1}^z & \Delta z_l^0 & \Delta z_{l+1}^0 \end{pmatrix}$$

$$+\det\begin{pmatrix}\Delta x^0_{l-1} & \Delta \boldsymbol{j}^x_l & \Delta x^0_{l+1}\\ \Delta y^0_{l-1} & \Delta \boldsymbol{j}^y_l & \Delta y^0_{l+1}\\ \Delta z^0_{l-1} & \Delta \boldsymbol{j}^z_l & \Delta z^0_{l+1}\end{pmatrix}$$

$$+\det\begin{pmatrix}\Delta x^0_{l-1} & \Delta x^0_l & \Delta \boldsymbol{j}^x_{l+1}\\ \Delta y^0_{l-1} & \Delta y^0_l & \Delta \boldsymbol{j}^y_{l+1}\\ \Delta z^0_{l-1} & \Delta z^0_l & \Delta \boldsymbol{j}^y_{l+1}\end{pmatrix},$$

we have a linear system

$$\boldsymbol{H}_l\ \Delta\boldsymbol{\rho} + \Delta X_l = 0. \tag{12}$$

From the above derivation, we see that every four consecutive touch points will decide a constraint Eq. (12). Using $(m+3)$ consecutive touch points, a linear system consisting of m constraint equations is obtained

$$\boldsymbol{H}\ \Delta\boldsymbol{\rho} + \Delta\boldsymbol{X} = 0,$$
$$\text{where } \boldsymbol{H} = [\boldsymbol{H}_1, \boldsymbol{H}_2, \ldots, \boldsymbol{H}_m]^{\mathrm{T}}$$
$$\text{and } \Delta\boldsymbol{X} = [\Delta X_1, \Delta X_2, \ldots, \Delta X_m]^{\mathrm{T}}. \tag{13}$$

In Eq. (13), the coefficient matrix $\boldsymbol{H}$ and $\Delta\boldsymbol{X}$ can be calculated based on the *difference* of nominal positions predicted by the controller, and the difference of special Jacobian for each pair of consecutive touch points (only joint readings are required for the computations). The only unknown remaining is the kinematic error to be identified. Eq. (13) serves as the linear identification model of the new method.

The derivation of the above model assumes the general case in which neither the position nor the orientation of the constraint plane is known accurately. As a special case, assuming that we have the knowledge of the orientation of the constraint plane with respect to the robot base co-ordinate system, for example, the constraint plane is aligned with the robot base x–y plane, then the z component of $\Delta\boldsymbol{p}_l$ will be zero. From Eq. (6), we have

$$\Delta \boldsymbol{j}^z_l\ \Delta\boldsymbol{\rho} + \Delta z^0_l = 0. \tag{14}$$

Comparing the above equation with the constraint Eq. (12), we see that the coefficient matrix $\Delta\boldsymbol{X}_l$ is simply the difference of z component of two consecutive touch points, and the $\boldsymbol{H}_l$ simply the difference of z component of special Jacobian at two consecutive touch points. Similarly, we have the constraint equations for the cases where the constraint plane is aligned with the robot base y–z or the x–z planes

$$\Delta \boldsymbol{j}^x_l\ \Delta\boldsymbol{\rho} + \Delta x^0_l = 0 \tag{15}$$

or

$$\Delta \boldsymbol{j}^y_l\ \Delta\boldsymbol{\rho} + \Delta y^0_l = 0. \tag{16}$$

Although the exact location of the constraint plane is not necessarily known, care must be exercised in placing the constraint plane in the workspace. Considerations include:

- the robot configurations enabling desirable and safe touch on the plane;
- the workspace in which accuracy is critical;
- the optimal identification configurations of the robot;
- the workcell layout and kind of constraint plane available.

As robot base axes are always set up to align with respect to some reference planes in the workcell, the assumption made earlier, in which the constraint plane is aligned with the robot base co-ordinates, has practical significance. In the case that external constraint planes are used, it is also easy to align the plane with the robot base axis using the probe and VAL II [1] axis motion function. To maximise the range of robot movements, it is desirable to have the robot touch the constraint planes separately which lie perpendicular to the robot's base axes. The linear identification model in this case can be constructed according to Eqs. (14)–(16). The calculations are simplified and the calibration results can be evaluated directly in the special case.

The resolution of Eq. (13) is equivalent to the minimisation of the following cost function in the least squares sense

$$E = \tfrac{1}{2}(\|\boldsymbol{H}\ \Delta\boldsymbol{\rho} + \Delta\boldsymbol{X}\|^2 + \alpha\|\Delta\boldsymbol{\rho}\|^2), \tag{17}$$

where α is a positive scalar coefficient for regulation [2] and $\|\ \|$ is a Euclidean norm. An extra regulation (penalty) term is added to improve the conditioning of identification Jacobian, which is near singular due to the partial pose information used in this case, and the limited movement ranges for data collection. The physical meaning of the regulation (penalty) term is to ensure a small norm of the identified kinematic error vector. Eq. (17) can be solved by using a standard least squares optimisation approach [11], or by using a real-time optimisation technique – recurrent neural network processing [16,19,2].

[1] For UNIMATION PUMA robot in this case.

3. Data Collection

Renishaw Probes were originally used for accurate workpiece set-up and workpiece measurement for CNC lathes. The probe is in effect an omni-directional switch that triggers when the probe contacts the workpiece from any direction [13]. For our application, a special tool changer was made to hold the probe (Fig. 2). The trigger signal is transmitted as an input to control the robot. The switch is kept on while the probe is in contact with the object. The tip-point of the probe is a ruby ball so that the contact point from any direction is a constant distance from the centre of the tip. The probe has 12.5° over-travel in $\pm x, y$ direction (which is equivalent to about 22 mm over-travel for a 100 mm long probe stylus), and 6.5 mm over-travel in the z direction, which allows a certain probing speed and misplacement of workpieces. The trigger force in the x, y direction is set at about 10 and 15 g in the z direction. The probe has repeatability of 1 μm and thus can be used for high precision measurements. The data collection procedure written in pseudo-code is as follows:

```
Repeat until the maximum number of touch points is reached {
        MoveTo(START);
        Point = RandomConfiguration(MAX, MIN);
        MoveTo(Point); while(ProbeSignal == OFF){
                ProbeBy(XSTEP, YSTEP, ZSTEP);
                }
        while(ProbeSignal == ON){
                ProbeBy(-XSTEP/10, -YSTEP/10, -ZSTEP/10);
                }
        RecordData(JointValues, CartesianValues);
        }
```

The robot moves from a start point to a point above the constraint plane where its configuration is randomly generated within the robot movement ranges, from which a desirable and safe touch on the constraint plane is ensured.[2] From that point, the robot probes the plane by moving in small steps toward the constraint plane until the probe signal is on. As there is some over-travel of the stylus, the stop point is not the point of the first touch due to the probing speed. Therefore fine tuning is needed to retract the first touch point. Whilst still in contact with the plane, the fine tuning process begins by moving the tip-point away from the plane in steps of one tenth of the probing steps. Then the joint values of the robot and the corresponding Cartesian co-ordinates are recorded for post-processing. The above process is repeated until the desired number of touch points has been reached.

The probing and fine tuning direction is along the normal to the constraint plane. The probing steps are normally set less than 1 mm hence the fine tuning steps are less than 0.1 mm, thus achieving a measurement accuracy in the order of 0.1 mm which is sufficient for robot calibration (if more accurate measurement is required, smaller probing steps can be set). In the special case when the constraint planes are aligned with robot base planes, the robot only probes in one axis direction, the movements along the other two axis directions are set to zero.

[2] This can be done either on-line using VAL II random generator, given the intervals of Cartesian co-ordinates, or through off-line planning process using robot simulation package.

4. Results for a PUMA 560 Robot

The PUMA 560 robot is a six-DOF manipulator with six revolute joints. There are in total 24 kinematic parameters, using the D-H notation, to describe the kinematic model. Since joints 2 and 3 of the PUMA robot are two consecutive joints with parallel joint axes, a modified D-H model is used to avoid model singularity [17]. The common normal parameter error

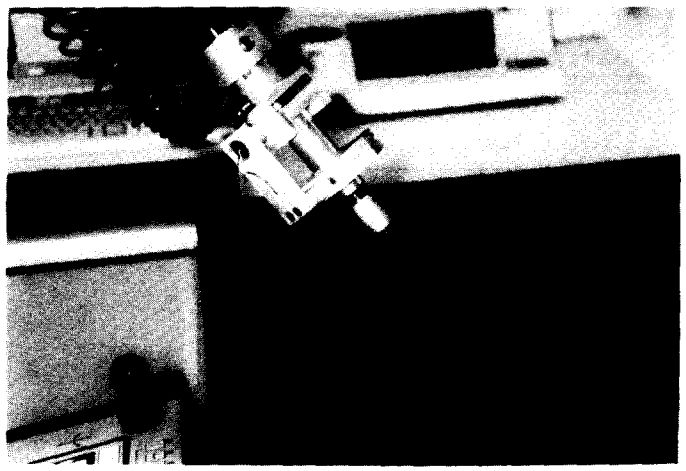


Fig. 2. Robot touch constraint plane using a *Probe*.

Table 1
Nominal parameters of a PUMA 560 robot

No.	a_i (mm)	d_i (mm)	α_i (rad)	β_i (rad)	TOOL	
1	0	0	$-\pi/2$	0	0.0	(rad)
2	431.8	149.09	0	0.0	0.0	(rad)
3	−20.32	0	$\pi/2$	0	0.0	(rad)
4	0	433.07	$-\pi/2$	0	5	(mm)
5	0	0	$\pi/2$	0	0.0	(mm)
6	0	56.25	0	0	302.34	(mm)

Δd_2 is replaced by a rotation parameter $\Delta\beta_2$, and the value of Δd_2 and the values of $\Delta\beta_i (i \neq 2)$ are fixed to zero during the identification process. In general, another six parameters including three each of rotation {roll–pitch–yaw} and translation parameters are needed for TOOL transformation. The nominal parameters for the PUMA robot are listed in Table 1. Since only position information is used for calibration, only three position parameters of the TOOL transformation are identifiable. To eliminate parameter redundancy between the TOOL and the last link, the TOOL parameters are incorporated into the last link. The three positional parameters of the TOOL are represented by the three parameters d_6, a_6 and θ_6 of the last link, and α_6 is not identifiable. The total number of identifiable kinematic parameters is therefore 23.

4.1. Simulation results

The simulation program was built to test the proposed calibration method. The flow chart of the simulation program was shown in Fig. 3. To maximise the robot joint movements, three constraint planes were simulated to be placed perpendicular to each base axis: $x = -550$ mm; $y = 300$ mm; $z = -450$ mm. Sixty random configurations in Cartesian space for each constraint plane were generated satisfying the constraint conditions. Then the nominal inverse kinematic model was used to find the corresponding joint values for each of the Cartesian configurations.[3] If there was no error in the kinematic model, it would be found that the positions achieved by controlling those joint values would perfectly match those constraint conditions. However, by inducing small errors in the parameters, the achieved positions by the 'actual' robot will be different from the commanded ones thus deviating from the constraint planes. An iterative inverse Jacobian method was adopted to find the corresponding joint compensations so that the positions achieved by the 'actual' robot were identical to the commanded ones. The updated joint values then feed to the nominal forward kinematic model to simulate the positions reported by the robot controller. The positions reported by the nominal forward kinematic model will not satisfy the plane constraint conditions. These position values, together with the robot special Jacobian, are used to calculate coefficient matrix $\boldsymbol{H}$ and $\Delta\boldsymbol{X}$ in Eq. (13). Then the recurrent neural network is applied to identify the kinematic errors [19,2]. The identified

[3] One of the eight robot joint configurations – RIGHT and ABOVE arm, and UP wrist – was selected.

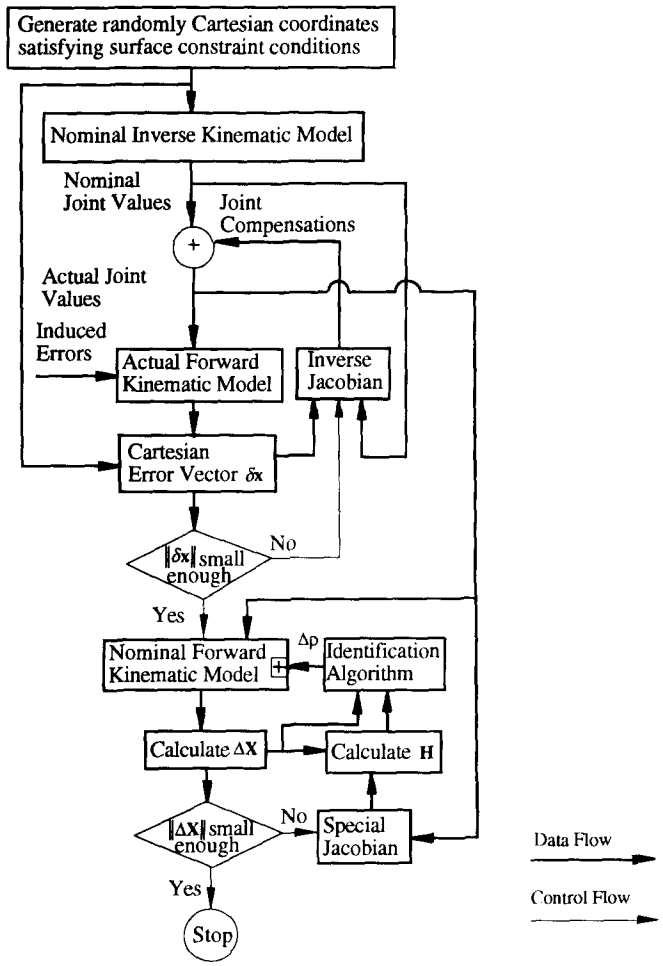


Fig. 3. Simulation program flow chart.

errors are then used to update the forward kinematic model. This process is repeated until the discrepancy is decreased to the desirable level. Finally the identified errors are compared with the induced errors to evaluate the simulation results.

Table 2 lists the identified kinematic errors using the identification model where neither orientation nor position of the constraint planes are known. The induced errors were randomly produced in the range of ± 0.15 mm for linear parameters and in the range of ± 0.015 rad for rotary parameters (the values in parenthesis). Comparing the induced errors with the identified, we can see that the angular errors are almost identical and the linear errors are similar but have small residual errors. Table 3 compares the deviations from the constraint planes before and after calibration. dx represents the deviations from the x-axis constraint plane $x = -550$ mm, dy represents the deviations from the y-axis constraint plane $y = 300$ mm, and dz represents the deviations from the z-axis constraint plane $z = -450$ mm. The statistical analysis is based on the 40 calibration points on each of the constraint planes. It is seen that the induced errors, though small, will produce a maximum deviation from the constraint plane of up to 13.2 mm. After identifying the kinematic errors, the deviations from the constraints planes are close to zero. Small residual errors exist in the x- and z-axes constraint planes (dx and dz), but the standard deviations of the residual errors are very small, which implies that the calibrated posi-

tions shift parallel from the constraint plane by a small amount. Increasing the magnitude of the induced errors will result in an invalid identification which leads to the identified model pointing to a plane parallel to the constraint plane (the constraint condition still holds). This is due to the fact that no information of the relative position and orientation between the constraint planes and robot base frame are provided in the identification. The general model is suitable for robot on-site re-calibration on a periodical basis where the parameter changes from the previous calibration are relatively small. Information about the relative relation between the constraint planes and robot base (such as the constraint planes aligned with the base) will enable the model to identify larger kinematic errors.

Table 2
Induced and identified kinematic errors

Link No.	Δa_i (mm)	Δd_i (mm)	$\Delta\alpha_i$ (rad)	$\Delta\theta_i$ (rad)	$\Delta\beta_i$ (rad)
1	0.087	0.055	0.0081	−0.0011	0
(induced)	(0.087)	(0.000)	(0.0080)	(−0.0019)	(0)
2	0.049	0	−0.0026	0.0059	0.0131
(induced)	0.069	(0)	(−0.0027)	(0.0060)	(0.0131)
3	−0.036	0.075	−0.0122	−0.0039	0
(induced)	(−0.004)	(0.011)	(−0.0123)	(−0.0039)	(0)
4	−0.061	−0.147	−0.0014	−0.0070	0
(induced)	(−0.053)	(−0.07)	(−0.0014)	(−0.0070)	(0)
5	0.015	0.029	0.0022	−0.0074	0
(induced)	(0.141)	(0.016)	(0.0022)	(−0.0076)	(0)
6	0.146	−0.039	0	−0.0084	0
(induced)	(0.146)	(0.040)	(0)	(−0.0070)	(0)

Table 3
Accuracy comparisons for calibration points

(mm)	Before calibration			After calibration		
	dx	dy	dz	dx	dy	dz
avg.	−3.756	−6.8745	−6.2575	−0.2049	−0.0876	−0.2646
stdev.	2.6299	2.7775	2.6459	0.0411	0.0429	0.0132
max.	6.5839	12.7363	13.1583	0.2681	0.1916	0.3098

Table 4
Induced and identified kinematic errors

Link No.	Δa_i (mm)	Δd_i (mm)	$\Delta\alpha_i$ (rad)	$\Delta\theta_i$ (rad)	$\Delta\beta_i$ (rad)
1	1.737	0.038	−0.0036	−0.0119	0
(induced)	(1.696)	(0.130)	(−0.0036)	(−0.0120)	(0)
2	1.175	0	−0.0012	−0.0175	0.0106
(induced)	(1.256)	(0)	(−0.0011)	(−0.0175)	(0.0106)
3	0.248	0.373	0.0160	−0.0005	0
(induced)	(0.236)	(0.374)	(0.0160)	(−0.0005)	(0)
4	0.179	1.560	−0.0106	−0.0048	0
(induced)	(0.195)	(1.662)	(−0.0106)	(−0.0048)	(0)
5	−0.788	−0.167	−0.0096	0.0076	0
(induced)	(−0.955)	(−1.367)	(−0.0130)	(0.0080)	(0)
6	−1.893	−1.932	0	0.0147	0
(induced)	(−1.894)	(−1.835)	(0)	(0.0138)	(0)

Table 4 lists the identification results using the identification model where the orientations of the constraint planes are known (aligned with the robot base axes). The induced errors were randomly produced in the range of ± 2 mm for linear parameters and in the range of ± 0.02 rad for angular parameters (the values in parenthesis). Both the linear and angular parameter errors identified are almost identical to those induced in this case. Comparing with the assumed positions of the constraint planes, the corresponding position component of the calibration points can be evaluated directly, which is listed in Table 5. The x component evaluation (dx) are based on the forty calibration points which lie on the constraint plane

Table 5
Accuracy comparisons for calibration points

(mm)	Before calibration			After calibration		
	dx	dy	dz	dx	dy	dz
avg.	−5.0646	1.9947	8.1754	0.1114	−0.0648	0.0051
stdev.	3.8930	3.6056	2.3923	0.0034	0.0093	0.0053
max.	14.8847	10.0196	13.0042	0.1186	0.1110	0.0148

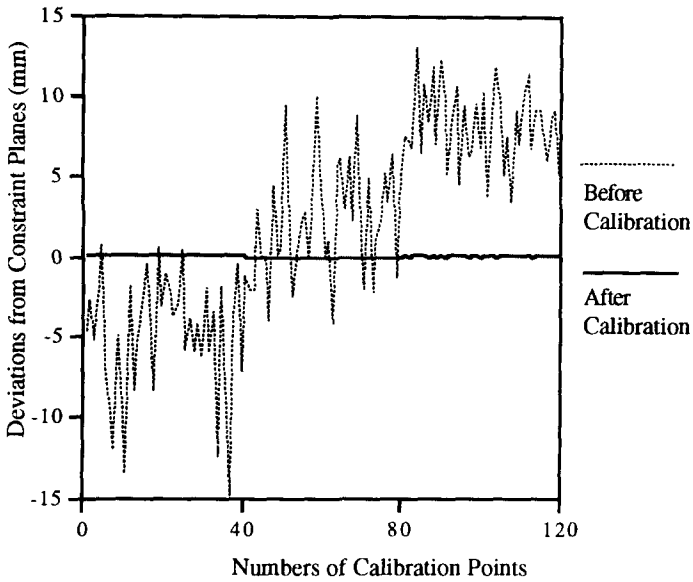


Fig. 4. Simulation result with induced errors.

$x = -550$ mm. The y, z component of position evaluation (dy and dz) are based on the calibration points which lie on the constraint plane $y = 300$ mm, and $z = -450$ mm, respectively. It is shown that the positioning accuracy has been improved significantly by identifying the induced errors, the maximum deviation from the constraint plane being decreased from up to 15 mm to below 0.2 mm.

Fig. 4 plots the x, y and z components of the positioning deviations from the corresponding constraint plane. The first 40 points are the x component deviations from the constraint plane $x = -550$ mm, the next 40 are the y component deviations from $y = 300$ mm constraint plane and the last 40 are the z component deviations from $z = -450$ mm constraint plane for the 40 calibration points on the corresponding constraint plane, respectively. Note that only relative position information (the nominal position difference between two consecutive touch points) is used for calibration. But the calibrated model can predict accurately the absolute positions of the constraint planes. There exists a small residual error in the x component deviations from the x-axis constraint plane after calibration (the average error is 0.1114 mm, Table 5). It is due to the fact that the linear position information between robot base and the constraint planes are not provided in the identification. The big value of induced errors used in the simulation resulted in the calibrated model pointing to the positions shifted parallel from the constraint plane. This small parallel shift will be diminished by decreasing the induced errors. Since the assumed errors are larger than those for actual robots we generally dealt with on-site calibration; the identification algorithm used in simulation is suitable for practical applications.

4.2. Experimental results

To verify the proposed calibration scheme, the experimental set-up was used for data collection (Fig. 5).

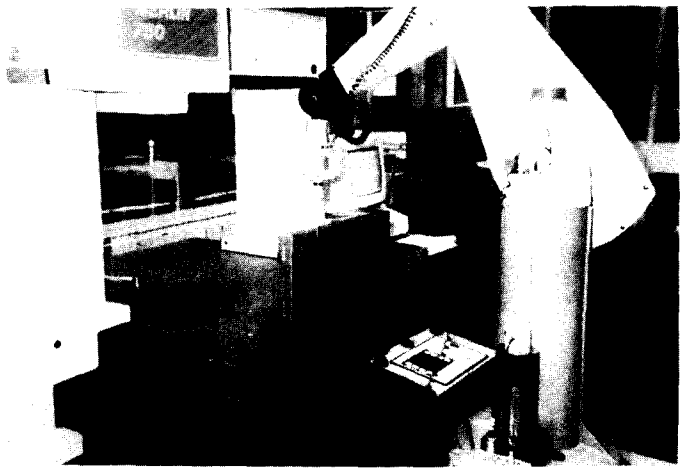


Fig. 5. Experimental set-up for data collection. The robot touches an aligned (x-axis) constraint plane using a probe.

A calibrated flat plate measuring 530 mm long and 250 mm wide was placed perpendicular to robot base x, y-axes for data collection. The robot x–y base plane was aligned precisely with the granite worktable of a co-ordinate measuring machine (CMM), therefore the worktable surface was used as the z-axis constraint plane. The positions of the x, y, z-axes constraint planes in the robot base frame, according to average values of the robot controller's readings, are at $x = -652.070$ mm; $y = 491.337$ mm; and $z = -470.558$ mm. The worktable surface has a reachable area for the robot of about $80 \times 110\,\text{cm}^2$ which allows a wide range of robot movements in the x, y directions. The robot movement ranges are restricted such that the probe touch points lie on the constraint planes. The flat plate has a flatness of about ± 0.001 mm and the flatness of the granite worktable surface is in the order of 5 μm which are accurate enough for robot calibration. The data collection procedure was implemented in VAL-II and it took about 5 s to collect one data point (the time needed depends on the speed of robot movement). One hundred touch points on each of the planes were collected. Sixty pairs randomly chosen from each of the 100 pairs were used for calibration and the remaining points were used for independent test. Table 6 lists the identified kinematic errors based on the experimental data.

Table 6
Identified errors of a PUMA 560 robot

Link No.	Δa_i (mm)	Δd_i (mm)	$\Delta\alpha_i$ (rad)	$\Delta\theta_i$ (rad)	$\Delta\beta_i$ (rad)
1	1.027	−0.010	0.0018	0.0017	0
2	0.688	0	−0.0023	0.0048	0
3	0.453	0.128	0.0007	0.0024	0
4	0.139	−0.252	0.0056	0.0172	0
5	−0.126	−0.441	−0.0061	0.0041	0
6	0.068	−0.348	0	0.0082	0

Fig. 6 plots the x, y and z components of the differences between each of the two consecutive touch points on the same constraint plane (the first 39 points are the difference of x components obtained from the 40 test points on the x-axis constraint plane, and the next 39 are y component differences obtained from the 40 test points on the y-axis plane and the last 39 are z component differences obtained from the 40 test points on the z-axis plane). The dashed lines represent the differences of positions predicted by the uncalibrated model in the robot controller while the solid lines represent the differences of positions predicted by the updated model using the identified errors. The symmetry of the graph is due to the use of the differences between consecutive points. It shows that the calibrated model works well for test data points as well. Therefore the calibrated model is valid not only for the calibration points but also for the test points.

Using the reference positions of the constraint planes perpendicular to the base axis, we can evaluate directly the positioning accuracy achieved by this calibration. The accuracy comparisons based on the forty test data points on each of the three constraint planes are given in Table 7, where dx, dy and dz represent

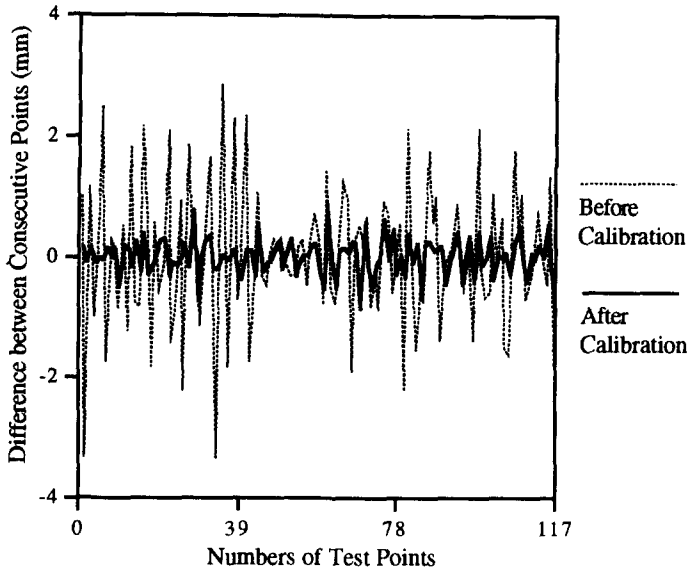


Fig. 6. Test result with experimental data.

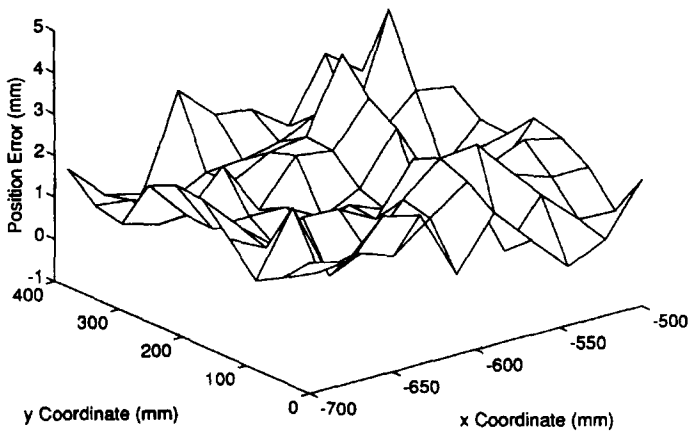
the x, y and z component deviations from the x, y and z-axes constraint plane, respectively. The average of absolute error after calibration has improved to below 0.3 mm. The z component deviations from the x–y plane (z-axis constraint plane) before and after this calibration are illustrated in Fig. 7 based on the 100 collected data from the z-axis constraint plane. The x, y-axes represent the x, y co-ordinates of those touch points on the z-axis constraint plane which shows that the touch points on the z-axis constraint plane lie in the area of $400 \times 200\,\text{mm}^2$, and the z-axis represents the z co-ordinate differences between the model predicted and the actual position of the z-axis constraint plane. Fig. 7(a) shows the z-axis constraint plane predicted by the robot model before calibration, while Fig. 7(b) illustrates the z-constraint plane predicted by the model after calibration. Fig. 7(b) is much closer to the actual shape and position of the z-axis constraint plane than Fig. 7(a). It shows that the deviations from the z-axis constraint plane has been decreased significantly after calibration. We can see that this approach achieves an accuracy improvement comparable to other calibration methods using sophisticated external measurements. The maximum inaccuracy after calibration is of the order of robot repeatability for the test points on the constraint planes (Table 7).

Table 7
Accuracy comparisons based on test points

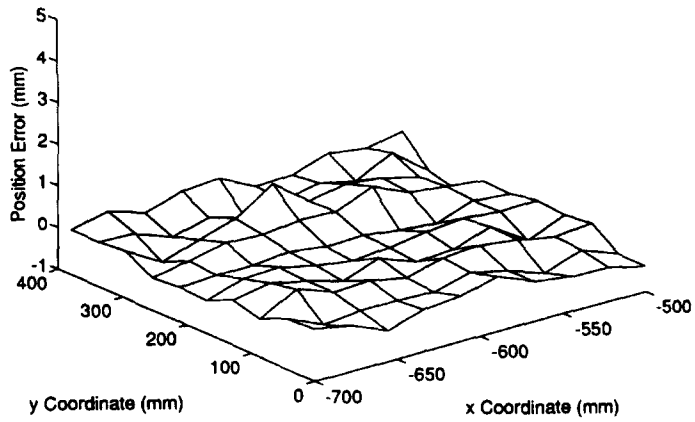
	Before calibration			After calibration		
(mm)	dx	dy	dz	dx	dy	dz
avg.	1.017	4.058	1.890	0.190	0.259	0.223
stdev.	0.165	0.658	0.306	0.031	0.042	0.036
max.	2.218	4.866	3.924	0.558	0.573	0.571

4.3. Cross-evaluation using a CMM

In the above sections, we have shown through simulation and experimentation that robot can calibrate its kinematic control model using the measurements of its internal sensors only. However, for the evaluation of the calibration results, the internal sensor measurements are inappropriate and some external global measurements are needed. Furthermore, although we show through experiment that the positioning accuracy of the test points on the constraint planes are improved, it needs to be verified that the positioning accuracy of those points beyond the constraint plane are also improved. Therefore a precision CMM was used to obtain the actual locations achieved by the robot end-effector. A measuring cube was mounted in the end-effector to obtain the full pose information of the end-effector. Two hundred and eighty eight points uniformly distributed in a vol-



(a). z-axis Constraint Plane Perceived by the Un-calibrated Robot Model



(b) z-axis Constraint Plane Perceived by the Calibrated Robot Model

Fig. 7. The z-axis constraint plane perceived by the robot: (a) Un-calibrated model; (b) Calibrated model.

ume of $200 \times 400 \times 200\,\text{mm}^3$ in Cartesian space and $45^\circ \times 90^\circ \times 135^\circ$ in orientation space were collected. The measurement volume is located near the area where the constraint plates were placed. Since the actual measurements of the CMM were with respect to a reference point, the reference point was calibrated before the collected data were used. The calibration of the reference point could be considered as the robot base calibration and was performed by minimising the total inaccuracy of the whole 288 points in the least square sense. To evaluate the robot kinematic model before and after calibration, the effects of the robot base and the tool were eliminated, since the robot was set up at different time for calibration and for evaluation and used the different end-effector tools.

Both the actual measured positions and orientations of the end-effector were compared with the locations reported by the kinematic model without calibration, and by the kinematic model updated using the identified errors of Table 5. The comparison results were listed in Table 8 which was based on twenty test points randomly chosen from the whole data set. After compensating robot base error, the average position error in length was decreased from 3.75 to 2.15 mm. The average error was decreased further from 2.15 to 0.76 mm after robot kinematic calibration. The standard deviation was decreased

Table 8
Cross-evaluation results using a CMM

	Before calibration		After base calibration		After base and robot calibration	
In length	Position	Orientation	Position	Orientation	Position	Orientation
avg.	3.7509	2.5743	2.15	1.57	0.76	0.84
stdev.	1.1506	0.4227	0.52	0.42	0.19	0.37
max.	4.7437	3.1496	2.86	2.45	1.14	1.41

Position in mm and orientation in degrees.

from 1.15 to 0.19. Although only position information was used in calibration, the orientation accuracy was improved as well due to the robot wrist parameters that were calibrated using the tool offset. The accuracy improvement of the cross-evaluation is encouraging, considering that the set up for calibration and evaluation was changed and the evaluation volume was beyond the constraint planes used for calibration.

5. Conclusions and Discussions

A new autonomous robot calibration scheme has been developed in this paper. Renishaw probes were originally used for workpiece set-up and measurements for CNC lathes. We applied these cost effective sensors successfully for robot on-site calibration in an industrial application environment. Instead of taking partial or complete pose measurements for robot calibration, the tip-point of the probe was constrained to a plane movement and only robot internal joint measurements were used for kinematic identification. Neither external measurements nor accurate fixture set-up are needed for such a calibration. The recurrent neural network-based parameter identification algorithm is used for calibration processing. Both simulation and experimental results for a PUMA robot show that robot positioning accuracy can be improved to the level of robot repeatability.

The six dimensional robot kinematic error model was projected into one positional dimension in this study. The full kinematic parameters can be calibrated as long as the inaccuracy function of the specific dimension component contains all the kinematic parameters. If the specific component is not sufficient to identify all kinematic parameters, more than one constraint plane can be placed in different positional axis separately for the data collection. Proper tool (probe) offsets are also needed to make the robot wrist parameters identifiable. The general kinematic identification model does not require the exact knowledge of the constraint plane locations but the model can only identify small kinematic errors. The orientation knowledge of the constraint planes (aligned with robot base planes) enable the model to identify reasonably large kinematic errors for practical applications. The alignment of the constraint plane with the robot base axis can be easily performed with the robot and a trigger probe. Future work will investigate the optimal placement of the constraint plane in the constrained environments so that the robot has the optimal identification configurations. The study could lead to the construction of a portable mechanical fixture for robot on-site calibration. Constraint surfaces other than planes may also be suitable for the proposed calibration method (such as a spherical ball) as long as the surfaces have known shapes and are suitable for touching with the probe. The concept of using a known shape reference object for robot calibration developed in this work can also be extended for non-contact type sensors such as CCD camera. The effects of measurement noises such as the flatness of constraint planes on identification accuracy need to be studied in future work.

Acknowledgements

The first author would like to acknowledge the CVCP and Napier University for financial support for this research. Our thanks are due to Mr. Bill Campbell for his contributions during the experiments, to Dr. Barry Keepence, Dr. Mike Mannion for reading of the manuscripts, and to Prof. A.R. Young for his consistent encouragement during the research. We are

also grateful to the anonymous reviewers for helpful comments.

References

[1] D.J. Bennett, D. Geiger and J.M. Hollerbach, Autonomous Robot calibration for hand–eye coordination, *International Journal of Robotics Research* 10 (5) (1991) 550–559.

[2] A. Cichocki and R. Unbehauen, *Neural Networks for Optimization and Signal Processing* (Wiley, New York, 1993).

[3] M.R. Driels and W. Swayze, Automated partial pose measurement system for manipulator calibration experiments, *IEEE Transactions on Robotics and Automation* 10 (4) (1994) 430–440.

[4] M.R. Driels, L.W. Swayze and L.S. Potter, Full-pose calibration of a robot manipulator using a coordinate measuring machine, *International Journal of Advanced Manufacturing Technology* 8 (1993) 34–41.

[5] K.S. Fu, R.C. Gonzalez and C.S.G. Lee, *Robotics: Control, Sensing, Vision, and Intelligence* (McGraw-Hill, New York, 1987).

[6] A. Goswami, A. Quaid and M. Peshkin, Complete parameter identification of a robot from partial pose information, *Proc. IEEE Int. Conf. on Robotics and Automation*, Vol.1 (1993) 168–173.

[7] R.P. Judd and A.B. Knasinski, A technique to calibrate industrial robots with experimental verification, *IEEE Transactions on Robotics and Automation* 6 (1) (1991) 20–30.

[8] C.R. Mirman and K.C. Gupta, Compensation of robot joint variables using special jacobian matrices, *Journal of Robotic System* 9 (1) (1992) 113–137.

[9] B.W. Mooring, Z.S. Roth and M.R. Driels, *Fundamentals of Manipulator Calibration* (Wiley, New York, 1991).

[10] W.S. Newman and D.W. Osborn, A new method for kinematic parameter calibration via laser line tracking, *Proc. IEEE Int. Conf. on Robotics and Automation*, Vol.2 (1993) 160–165.

[11] U.S. Pathre and M.R. Driels, Simulation experiments in parameter identification for robot calibration, *International Journal of Advanced Manufacturing Technology* 5 (2) (1990) 13–33.

[12] J.R. Prenninger, M. Vincze and H. Gander, Contactless position and orientation measurement of robot end-effector, *IEEE Int. Conf. Robotics and Automation*, Atlanta, Vol.1 (1993) 180–185.

[13] Renishaw Metrology Ltd., *User's Guide: LP2 and LP2H Probes with hard wired or induction transmission* (1983).

[14] H.W. Stone, A.C. Sanderson and C.P. Neuman, Arm signature identification, *Proc. IEEE Int. Conf. on Robotics and Automation*, San Francisco, CA (1986) 41–48.

[15] G.-R. Tang and B.W. Mooring, Plane-motion approach to manipulator calibration, *International Journal of Advanced Manufacturing Technology*, Vol.7 (1992) 21–28.

[16] D. Tank and D. Hopfield, Simple 'neural' optimisation networks: An A/D converter, signal decision circuit, and linear programming circuit, *IEEE Transactions on Circuits and System* CAS-33 (1986) 533–544.

[17] W.K. Veitschegger and C.-H. Wu, Robot calibration and compensation, *IEEE International Journal of Robotics and Automation* 4 (6) (1988) 643–656.

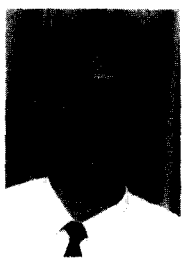
[18] D. Whitney, Lozinsky and J. Rourke, Industrial robot forward calibration method and results, *ASME Journal of Dynamic Systems, Measurement and Control* 108 (1986) 1–8.

[19] X.-L. Zhong and J.M. Lewis, Kinematic identification and compensation of robot manipulators using neural optimisation networks, *Proc. of the Third Int. Conf. on Automation, Robotics and Computer Vision*, Singapore, Vol. III (1994) 1472–1476.

[20] X.-L. Zhong, J.M. Lewis and F. N.-Nagy, *Robot Inverse Calibration Using Artificial Neural Networks*, (Pergamon Press, Oxford, 1995), submitted to *Journal of Engineering Applications of Artificial Intelligence*.

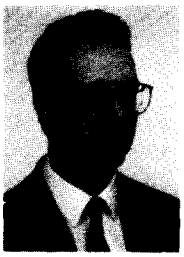
[21] X.-L. Zhong, J.M. Lewis and H. Rea, Neuro-accuracy compensator for industrial robots, *Proc. of IEEE Int. Conf. on Neural Networks (WCCI'94)* Orlando, Florida, Vol.5 (1994) 2797–2802.

[22] H. Zhuang, L. Wang and Z.S. Roth, Simultaneous calibration of a robot and a hand-mounted camera, *Proc. IEEE Int. Conf. on Robotics and Automation*, Vol.2 (1993) 149–154.



Xiao-Lin Zhong received the BSc degree in mechanical engineering in 1986 and the MSc degree in Robotics and Automation in 1989 from Huazhong University of Science and Technology (HUST), Wuhan, China and the PhD degree from Napier University, Edinburgh, UK in 1996. He joined Shenzhen Sciences and Industrial Park Corp, Shenzhen, China as a R&D engineer in 1989. He is currently a project manager with China Aerospace International Holdings Ltd. based in Hong Kong.

He was a major participant in the "Wuhan-I autonomous mobile robot system" project while working for his masters degree at HUST; mainly responsible for the development of microprocessor-based control system and the integration of multi-sensor systems for robot navigation. For which he was awarded an outstanding thesis prize from HUST in 1991. His research interests include robot autonomous calibration, motion planning and control, artificial neural network and its applications in an intelligent manufacturing system.



John M. Lewis received the BSc degree in Engineering Science from Exeter University, England in 1982. He joined Westland Helicopters Ltd as a systems engineer working on pilot decision support systems for rotor craft and in 1984 the company supported him through a PhD programme in artificial intelligence at Edinburgh University. In 1987 Dr. Lewis joined the Department of Mechanical, Manufacturing and Software Engineering at Napier University, Edinburgh where he is currently a senior lecturer in software engineering. His research interests include autonomous robot calibration, real time job scheduling for flexible assembly systems, and distributed requirements gathering techniques.

Prof. Francis L.N.-Nagy received the BSc degree in control engineering and a MSc degree in telecommunication from the University of Technical Sciences, Budapest, Hungary and then a PhD from the University of Salford (UK). He came to England in 1956 and became a Senior Research Engineer with Epsylon Ltd, and in 1959 was appointed Chief Engineer of Vectron Research Laboratories where he was involved in rocket research work with Professor Dennis Gabor, Nobel Laureate at Imperial College, London. In 1964, as a lecturer, he joined the Department of Electrical Engineering, University of Salford, where he lectured servomechanisms and advanced robot design on postgraduate courses. Later on he became senior lecturer and then Professor at the IT-Institute of Salford University. He has published three books in robotics and solid state control devices (in the UK and USA), more than hundred papers in automation and control applications. He has also a number of inventions patented in the UK and France. Professor N-Nagy is still with the IT-Institute and a consultant in industrial control and leading edge technologies in robotics. In 1986 he was associated with the establishment of UK National Advanced Robotics Centre at Salford and served on its Steering Committee. In 1993 he was one of the initiators for establishing the European Association for Research of Leg Robot in Brussels (EARLR) and became the Association's first Vice-President. He is a chartered engineer of the Institution of Electrical Engineers (UK), Fellow of IEE (UK), Fellow of the FIMA (Institute of Mathematics and Applications) (UK), Senior Member of IEEE (USA).