

Research Article

IMU-Based Online Kinematic Calibration of Robot Manipulator

Guanglong Du and Ping Zhang

School of Computer Science & Engineering, South China University of Technology, Guangzhou 510006, China

Correspondence should be addressed to Ping Zhang; pzhang_scut@163.com

Received 5 August 2013; Accepted 12 September 2013

Academic Editors: Y. Lu and M. Strojnik

Copyright © 2013 G. Du and P. Zhang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Robot calibration is a useful diagnostic method for improving the positioning accuracy in robot production and maintenance. An online robot self-calibration method based on inertial measurement unit (IMU) is presented in this paper. The method requires that the IMU is rigidly attached to the robot manipulator, which makes it possible to obtain the orientation of the manipulator with the orientation of the IMU in real time. This paper proposed an efficient approach which incorporates Factored Quaternion Algorithm (FQA) and Kalman Filter (KF) to estimate the orientation of the IMU. Then, an Extended Kalman Filter (EKF) is used to estimate kinematic parameter errors. Using this proposed orientation estimation method will result in improved reliability and accuracy in determining the orientation of the manipulator. Compared with the existing vision-based self-calibration methods, the great advantage of this method is that it does not need the complex steps, such as camera calibration, images capture, and corner detection, which make the robot calibration procedure more autonomous in a dynamic manufacturing environment. Experimental studies on a GOOGOL GRB3016 robot show that this method has better accuracy, convenience, and effectiveness than vision-based methods.

1. Introduction

Because of the manufacturing and assembly tolerance, the actual kinematic parameters of a robot deviate from their nominal values, which are referred to as kinematic errors. The kinematic errors would result in the errors of the robot tool if the nominal kinematics were used to estimate the pose of the robot. With the restriction of cost, the kinematic calibration is an effective way to improve the absolute accuracy of robots. Nowadays, calibration tasks use a lot of measurement techniques like coordinate measuring machines, laser tracking interferometer systems, and inexpensive customized fixtures [1, 2]. These systems are not only very expensive but also not friendly to use or with low working volume. A system which is used in a dynamic environment is expected to perform calibration without any external expensive calibration apparatus and elaborate setups, which means self-calibration.

Self-calibration techniques can be classified into two kinds: (1) redundant sensor approach and (2) motion constraint approach.

To increase the degrees-of-sensing over DOF, the redundant sensors approach includes one or more redundant rotary sensors to the proper passive joints of the manipulator.

There is a self-calibration method for parallel mechanisms with a case study on Stewart platform which is proposed by Zhuang in [3]. He used forward and inverse kinematics with six rotary encoders for three objective functions of parameter identification. Khalil and Besnard [4] installed two orthogonally allocated inclinometers to the tool to calibrate the Stewart platform except the redundant sensors which are mentioned above. However, there are some limitations of these methods. One of them is that some kinematic parameters orthogonally are not independent of the error models and the position and/or orientation of the tool on the platform cannot be calibrated.

For the other approach, that is the motion constraint approach, the mobility of the resultant system will be lower than its inherent degrees-of-sensing by fixing one or more passive joints or constraining partial DOF of the manipulator so that the calibration algorithm can be performed [5]. Bennett and Hollerbach [6] lowered the mobility of the tool of a serial manipulator and performed self-calibration using only the inherent joint sensors in the manipulator. And this idea was used and extended to calibrate a robot system with a hand-mounted instrumented stereo camera [7]. However, the position and/or orientation of the tool on the platform

cannot be calibrated, and some parameter errors related to the locked passive joints may become unobservable in the calibration algorithm because of the mobility constraints.

To solve these limitations, advances in robot calibration allow the researchers to use a hand-mounted camera to calibrate a robot instead of using measurements from passive joints or imposing mechanical constraints. Compared with those mechanical measuring devices, this camera system costs less and it is easier to use and more accurate. The traditional vision-based methods [8–15] to calibrate a robot require the precise 3D fixtures measured in a reference coordinate system and the procedure is inconvenient and time consuming and it may not be feasible for some applications. The self-calibration methods [16, 17] assume that the camera is rigidly attached to the robot tool. Closed-loop methods “virtual closed kinematic chain” proposed in [18–20], use the joint angle measurements already in the robot and can be considered self-calibrating. A method uses laser to capture robot position data to model the stiffness of the manipulator [21] and predict kinematic parameters [22–26]. O’Brian et al. [27] used a magnetic motion to capture robot data to estimate the kinematic parameters. Renaud et al. [28] and Rauf et al. [29] used a vision-based measuring device and a pose measurement device for kinematic calibration, respectively. Santolaria et al. [30] employed a continuous data capture method by using a ball bar gauge and a coupling probe to estimate the kinematic parameters. However, these approaches have a limitation; that is, the calibration is completed off-line. The optimization technique was based on the measured positions of the EE. The parameter error was minimized in the measured positions, but the error increased in very different positions. Moreover, the parameter error increased while the robot withstood different loads. When the robot is used in high-temperature or high-pressure environments, such as deep sea or outer space, the shapes of the robot links are easy to change. Therefore, online calibration is an indispensable method to rectify the kinematic parameters in real time.

In this paper, we propose an original approach of online robot calibration using IMU to measure the robot poses. In our method, an IMU is required to rigidly attach to the robot tool (Figure 1) to measure the robot pose in real time. In order to reduce the effect of the noise and improve the accuracy, we proposed a method combined FQA and KF to estimate the orientation of the IMU. Finally, an EKF is used to estimate differential errors of individual kinematic parameters. Unlike existing vision-based self-calibration methods, the described method does not require special complex steps such as camera calibration and corner detection. Moreover, this method does not require robot to make the motion for capturing the images, which makes our method more efficient.

The remainder of the paper is organized as follows. Section 2 provides kinematic modeling for the serial robot. In Section 3, a method of pose measurement using IMU is presented. Parameters identification algorithm is proposed in Section 4. In Section 5, an EKF is detailed to estimate the kinematic parameter errors. Finally, the experimental results are shown in Section 6 and we conclude the paper in Section 7.

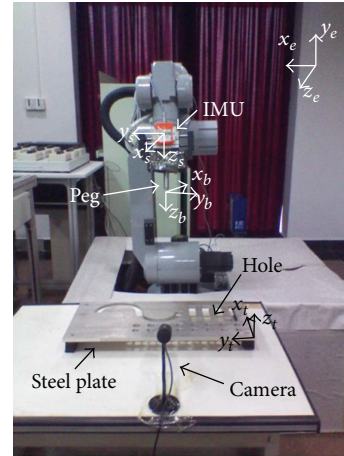


FIGURE 1: Structure of the system.

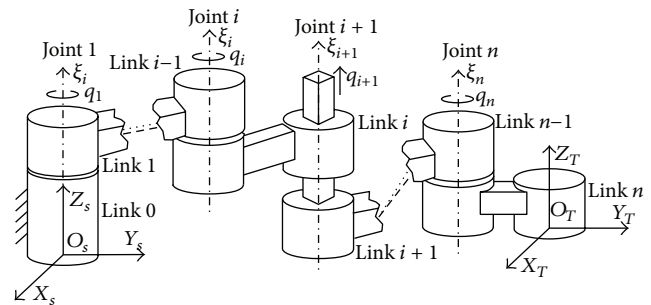


FIGURE 2: Forward kinematics of an n -DOF robot.

2. Kinematic Modeling

A robot kinematic model relates the robot joint coordinate to the pose of the robot tool. A robot kinematic model should meet the following rules for the kinematic parameter identification [21–23].

- (1) Completeness: the robot kinematic model should have enough parameters to define any possible deviation from the nominal values [24].
- (2) Continuity: any small changes in the structure of the robot must correspond to small changes in kinematic parameters [21].
- (3) Minimality: the kinematic model must include only a minimal number of parameters [3].

Many researchers have found suitable kinematic models for robot since 1980s, such as Hayati et al. models [25–27], Veitschegger and Wu’s model [28], Stone and Sanderson’s S-model [29], and Zhuang et al. model [30]. The standard Denavit-Hartenberg (DH) [31] convention is the most often used to describe the robot kinematics (Figure 2). The error models of DH are not continuous for robots that possess parallel joint axes. To avoid the singularity of DH convention, the DH modeling or Hayati modeling convention were used, respectively. The singularity-free calibration model prevents

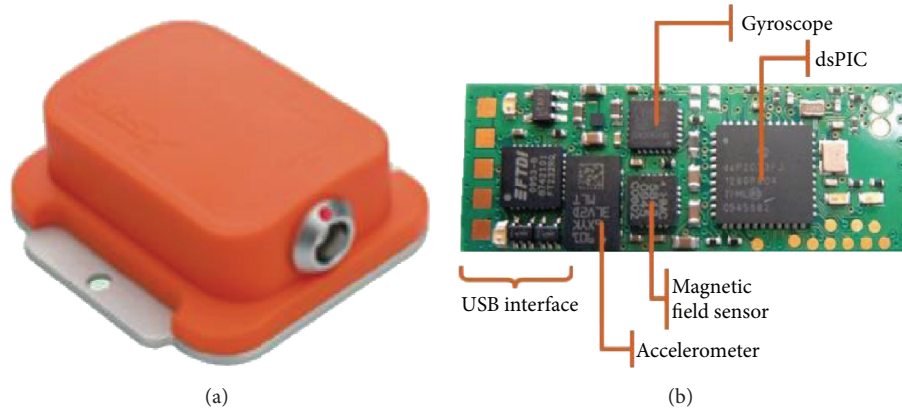


FIGURE 3: IMU sensor (a) and the prototype board (b).

the use of a single minimal modeling convention which can be used to identify all possible robot parameters.

The robot tool position and orientation are defined according to the controller conventions. Through the consecutive homogeneous transformations from the base coordinate to the robot tool coordinate (Figure 2), the kinematic equation can be defined as

$$T_N^0 = T_N^0(\mathbf{v}) = T_1^0 T_2^1 \dots T_N^{N-1} = \prod_{i=1}^N T_i^{i-1}, \quad (1)$$

where T_i^{i-1} is the translation matrix from $i - 1$ coordinate to i coordinate, N is the number of joints (or joints coordinates), $\mathbf{v} = [v_1^T v_2^T \dots v_N^T]^T$ is the parameter vector for the robot, and v_i^T is the link parameter vector, which includes the joint errors

$$v_i = v_i' + \Delta v_i, \quad (2)$$

where v_i' is the nominal vector for the joint i and $\Delta v = [\Delta v_1^T \Delta v_2^T \dots \Delta v_N^T]^T$ is the link parameter error vector for the joint i . The exact kinematic equation is

$$K_N^0 = K_N^0(\mathbf{v}) = K_1^0 K_2^1 \dots K_N^{N-1} = \prod_{i=1}^N K_i^{i-1}, \quad (3)$$

$$K_i^{i-1} = T_i^{i-1}(v_i' + \Delta v_i) = T_i^i + \Delta T_i. \quad (4)$$

Taking the joint variables into consideration, thus,

$$K_N^0 = T_N^0 + \Delta T, \quad \Delta T = \Delta T(\mathbf{u}, \Delta v), \quad (5)$$

where $\mathbf{u} = [\theta_1^T \theta_2^T \dots \theta_N^T]^T$ is a vector of joint variables. Thus, $K_N^0 : \mathfrak{R}^n \times \mathfrak{R}^N$ is a function of \mathbf{u} and Δv .

3. Pose Measurement Using IMU

3.1. Factored Quaternion Algorithm (FQA). The FQA presented in [32], which is based on Earth gravity and magnetic field measurements, is for estimating the orientation of a rigid body. This algorithm is only applied for the static or slow-moving rigid body. In order to be applicable to situations

in which relatively large linear accelerations occur, we use *KF* fusion algorithm together with angular rate information to estimate the orientation of (slow-moving or fast-moving) dynamic body in the next section.

In our application, a sensor module, which is a strap down inertial measurement unit (IMU) is attached to the robot tool whose orientation (roll, pitch, and yaw) is to be determined. Figure 3 shows an IMU of *Xsens* and its prototype board. The IMU sensor consists of one three-axis accelerometer, two two-axis gyroscopes, and one three-axis magnetometer.

We define three frames: body frame $x_b y_b z_b$, sensor frame $x_s y_s z_s$, and Earth-fixed frame $x_e y_e z_e$. The sensor frame $x_s y_s z_s$ corresponds to the axes of three orthogonally mounted accelerometers/magnetometers. Because the sensor is attached to the robot tool rigidly, the body frame $x_b y_b z_b$ is assumed to coincide with the sensor frame $x_s y_s z_s$. The Earth-fixed frame $x_e y_e z_e$ is defined to follow the North-East-down (NED) convention, where x_e points North, y_e points East, and z_e points down. The IMU can measure the orientation (roll, pitch, and yaw) of itself. Define that the rotation ϕ about the x_e -axis represents roll, the rotation θ about y_e -axis represents pitch, and the rotation ψ about z_e -axis represents yaw. According to Euler's theorem [33] on finite rotations, the conversion from Euler angles to quaternions is

$$q = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{\phi}{2}\right) \cos\left(\frac{\theta}{2}\right) \cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right) \sin\left(\frac{\theta}{2}\right) \sin\left(\frac{\psi}{2}\right) \\ \sin\left(\frac{\phi}{2}\right) \cos\left(\frac{\theta}{2}\right) \cos\left(\frac{\psi}{2}\right) - \cos\left(\frac{\phi}{2}\right) \sin\left(\frac{\theta}{2}\right) \sin\left(\frac{\psi}{2}\right) \\ \cos\left(\frac{\phi}{2}\right) \sin\left(\frac{\theta}{2}\right) \cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right) \cos\left(\frac{\theta}{2}\right) \sin\left(\frac{\psi}{2}\right) \\ \cos\left(\frac{\phi}{2}\right) \cos\left(\frac{\theta}{2}\right) \sin\left(\frac{\psi}{2}\right) - \sin\left(\frac{\phi}{2}\right) \sin\left(\frac{\theta}{2}\right) \cos\left(\frac{\psi}{2}\right) \end{bmatrix}, \quad (6)$$

and the four Euler parameters are constrained as [34]

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1, \quad (7)$$

where q_0 is the scalar part and (q_1, q_2, q_3) is the vector part. So the direction cosine matrix M_s^e from the sensor frame to the Earth-fixed frame can be represented as follows:

$$M_s^e = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_0q_2 + q_1q_3) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_0q_1 + q_2q_3) & q_0^2 - q_1^2 - q_2^2 - q_3^2 \end{bmatrix}. \quad (8)$$

3.2. Quaternion KF. Because both gyroscopes and magnetometer have white noise and random walk, we use Kalman Filter to estimate the state x of IMU from a set of noisy and incomplete measurements [35]. The Kalman Filter is a recursive stochastic technique and it estimates the state at time k from the state at time $k - 1$. The state-transition equation at time k is

$$\begin{aligned} x_k &= A_k \cdot x_{k-1} + B \cdot u_{k-1} + w_{k-1}, \\ z_k &= H \cdot x_k + v_k, \end{aligned} \quad (9)$$

where x_k is the $n \times 1$ state vector at time k , A is an $n \times n$ state-transition matrix, B is an $n \times p$ system input matrix, u_{k-1} is a $p \times 1$ vector with deterministic input at time $k - 1$, w_{k-1} is an $n \times 1$ process noise vector at time $k - 1$, z_k is an $m \times 1$ measurements vector at time k , H is an $m \times n$ observation matrix, and v_k is an $m \times 1$ measurement noise vector. In this paper, $n = 7$ and $m = 6$. The differential equation of the quaternion q with respect to time is

$$\begin{bmatrix} \frac{\partial q_0}{\partial t} \\ \frac{\partial q_1}{\partial t} \\ \frac{\partial q_2}{\partial t} \\ \frac{\partial q_3}{\partial t} \end{bmatrix} = \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ v_x \\ v_y \\ v_z \\ 2 \\ 2 \end{bmatrix}, \quad (10)$$

where v_x , v_y , and v_z are the angular velocity components of IMU in x_s -, y_s -, and z_s -axis. Since the state x_k includes the quaternion states and the angular velocities, x_k has the following form:

$$x_k = [q_{0,k} \quad q_{1,k} \quad q_{2,k} \quad q_{3,k} \quad v_{x,k} \quad v_{y,k} \quad v_{z,k}]^T, \quad (11)$$

where $q_{0,k}$, $q_{1,k}$, $q_{2,k}$, $q_{3,k}$, $v_{x,k}$, $v_{y,k}$, and $v_{z,k}$ are the quaternion states and the angular velocities at time k . From (10), the state-transition matrix is

$$A_k = \begin{bmatrix} 1 & 0 & 0 & 0 & -q_{1,k} \cdot \frac{\Delta t}{2} & -q_{2,k} \cdot \frac{\Delta t}{2} & -q_{3,k} \cdot \frac{\Delta t}{2} \\ 0 & 1 & 0 & 0 & q_{0,k} \cdot \frac{\Delta t}{2} & q_{3,k} \cdot \frac{\Delta t}{2} & q_{2,k} \cdot \frac{\Delta t}{2} \\ 0 & 0 & 1 & 0 & q_{3,k} \cdot \frac{\Delta t}{2} & q_{0,k} \cdot \frac{\Delta t}{2} & -q_{1,k} \cdot \frac{\Delta t}{2} \\ 0 & 0 & 0 & 1 & -q_{2,k} \cdot \frac{\Delta t}{2} & q_{1,k} \cdot \frac{\Delta t}{2} & q_{0,k} \cdot \frac{\Delta t}{2} \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (12)$$

where Δt is the sampling time. Let $B = 0^{n \times p}$ because there is no control inputs. We use angular velocities to estimate the quaternion states, so the process noise vector is

$$w_k = [0 \quad 0 \quad 0 \quad 0 \quad w_x \quad w_y \quad w_z]^T, \quad (13)$$

where w_x , w_y , and w_z are the process noises of the angular velocity. Because we use calibrated gyroscopes to measure the angular velocities, the observation matrix H is

$$H = [0^{3 \times 4} \quad I^{3 \times 3}]. \quad (14)$$

In order to satisfy (7), the determined quaternion q_k at time k should be normalized by

$$\begin{aligned} q_k &= \begin{bmatrix} q_{0,k} & q_{1,k} & q_{2,k} & q_{3,k} \\ M & M & M & M \end{bmatrix}^T, \\ M &= \sqrt{q_{0,k}^2 + q_{1,k}^2 + q_{2,k}^2 + q_{3,k}^2}. \end{aligned} \quad (15)$$

4. Parameter Identification

Kinematic identification is the process that identifies the kinematic model parameters of a robot manipulator by a given set of robot tool pose measurements and the corresponding joint position readings. The objective of a kinematic identification algorithm is to minimize the difference between the computed and the measured poses [15].

Assuming that the number of measured pose is m , it can be stated that

$$\begin{aligned} \widehat{K} &= \widehat{K}_N^0 = (\widehat{K}(\mathbf{u}_1, \mathbf{v}), \widehat{K}(\mathbf{u}_2, \mathbf{v}), \dots, \widehat{K}(\mathbf{u}_m, \mathbf{v}))^T, \\ \Delta \widehat{T} &= \Delta \widehat{T}_N^0 = (\Delta \widehat{T}(\mathbf{u}_1, \mathbf{v}), \Delta \widehat{T}(\mathbf{u}_2, \mathbf{v}), \dots, \Delta \widehat{T}(\mathbf{u}_m, \mathbf{v}))^T, \end{aligned} \quad (16)$$

where \mathbf{u}_i ($i = 1, 2, \dots, m$) is the vector of joint variables for the i measure pose.

All matrices or vectors in bold are functions of m . The objective of the kinematic identification is the computation for the parameter vector $\mathbf{v}^* = \mathbf{v}' + \Delta \mathbf{v}$, which is to minimize the discrepancy between the computed and the measured poses:

$$A(\mathbf{v}^*, \mathbf{u}) = B(\mathbf{u}) \quad (17)$$

A is the function of pose of \widehat{T} and $B(\mathbf{u}) = (B(u_1), B(u_2), \dots, B(u_m))^T$ is the measured function of joint variables \mathbf{u} .

For each measurement pose $B(u_i)$, it concludes orientation measurement $R_i \in \mathfrak{R}^{3 \times 3}$ and position measurement $P_i \in \mathfrak{R}^{3 \times 1}$, and

$$B(u_i) = \begin{bmatrix} R_i & P_i \\ \mathbf{0} & 1 \end{bmatrix}, \quad u_i \in \mathfrak{R}^{4 \times 4}. \quad (18)$$

If the measurement system can provide orientation measurement and position measurement, each pose can formulate six measurement equations. If only orientation measurement can be provided by the measurement system, each pose measurement can just formulate three measurement

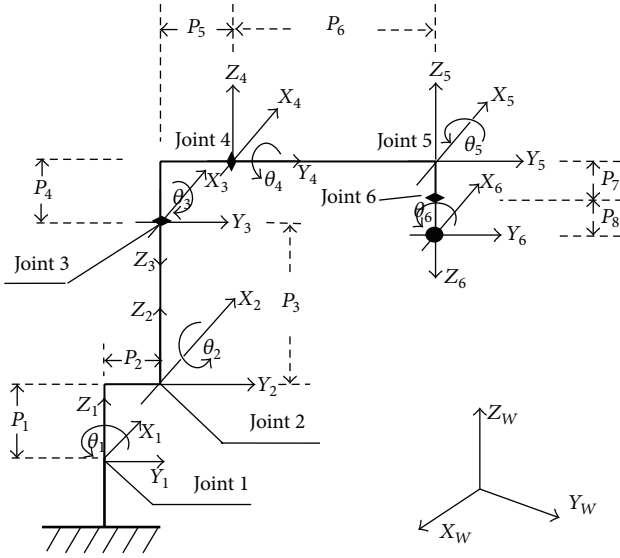


FIGURE 4: Skeleton of the GOOGOL GRB3016 robot with coordinate frames in the zero position.

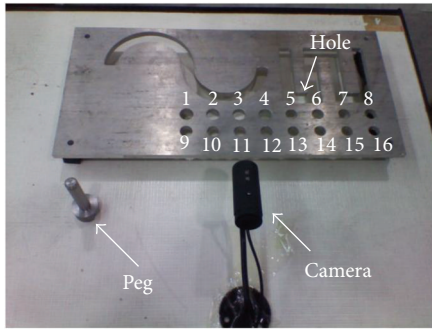


FIGURE 5: Steel plate and errors measurement system.

equations. In this paper, only orientation obtained from IMU is used to calibration the kinematic parameters. From (17),

$$A(v^*, \mathbf{u}) = B(\mathbf{u}) = A(v, \mathbf{u}) + C(\Delta v, \mathbf{u}), \quad (19)$$

where C is the discrepancy function of the orientation components of $\Delta \hat{T}$. Introducing the Jacobian matrix,

$$C(\Delta v, \mathbf{u}) = J \cdot \Delta v, \quad (20)$$

and then

$$C(\Delta v, \mathbf{u}) = B(\mathbf{u}) - A(v, \mathbf{u}), \quad (21)$$

when using

$$b = B(\mathbf{u}) - A(v, \mathbf{u}) \in \mathfrak{R}^{4 \times 4 \times m}, \quad (22)$$

$$x = \Delta v \in \mathfrak{R}^{4 \times 4 \times m}. \quad (23)$$

Equation (20) can be rewritten:

$$J \cdot x = b. \quad (24)$$

5. Estimating Errors Using Extended Kalman Filter

Initially, the orientations of the tool are measured from the IMU. Since uncertainty exists in the measurement, Extended Kalman Filter (EKF) is used as an optimization algorithm and the Jacobian matrices are used to estimate the kinematic errors of DH parameters by the measured orientation values [5].

Since there are four parameters for N revolute joints and four parameters for the transformation from the IMU to the tool, the number of total parameters to be considered is $4(N + 1)$. So the predicted state \hat{x} is $4(N + 1)$ of the DH parameters in the prediction step of the EKF. The covariance matrix of the predicted state P is

$$\begin{aligned} \hat{x}_{k+1|k} &= \hat{x}_{k|k}, \\ P_{k+1|k} &= P_{k|k} + Q_k, \end{aligned} \quad (25)$$

where Q_k is the covariance matrix of the system noise at the k th iteration.

In the observation step of the EKF, Jacobian matrix J (20), measurement residual \tilde{y} , and residual covariance S are calculated as follows:

$$\begin{aligned} J_{k+1} &= \left. \frac{\partial T(x)}{\partial x} \right|_{\hat{x}_{k+1|k}}, \\ \tilde{y}_{k+1} &= m_{k+1} - T(\hat{x}_{k+1|k}), \\ S_{k+1} &= J_{k+1} P_{k+1} J_{k+1}^T + R_{k+1}, \end{aligned} \quad (26)$$

where m_k and R_k are the measured orientation value and the covariance matrix of measurement noise at the k th iteration. $k + 1|k$ means a prior estimate, and $k + 1|k + 1$ means a posteriori estimate.

In the update step, the state covariance matrix is updated by an optimal Kalman gain K :

$$\begin{aligned} K_{k+1} &= P_{k+1|k} J_{k+1}^T S_{k+1}^{-1}, \\ \hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k} + K_{k+1} \tilde{y}_{k+1}, \\ P_{k+1|k+1} &= (I - K_{k+1} J_{k+1}) P_{k+1|k}, \end{aligned} \quad (27)$$

where I is the identity matrix. The norm values of the state vector are calculated for every iteration once the updating procedure is completed. Note that if Q and R are set to zero, then EKF simply reduces to the Newton-Raphson method.

6. Experimental Results

6.1. Experiments Environment. To verify the above method, a GOOGOL GRB3016 robot model was used in this experiment. In the experiment, the robot could self-calibrate online in the working status. There were four steps in the self-calibration procedure.

- (1) Data collection: the orientation of the IMU and the corresponding joint angles were captured with the robot tool moving in different pose.

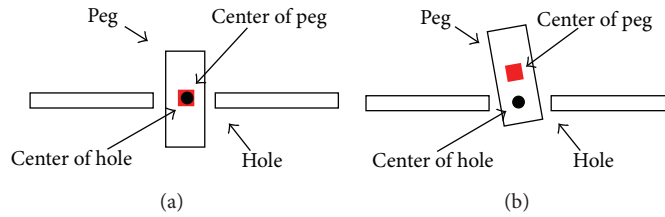


FIGURE 6: Definition of 3D errors.

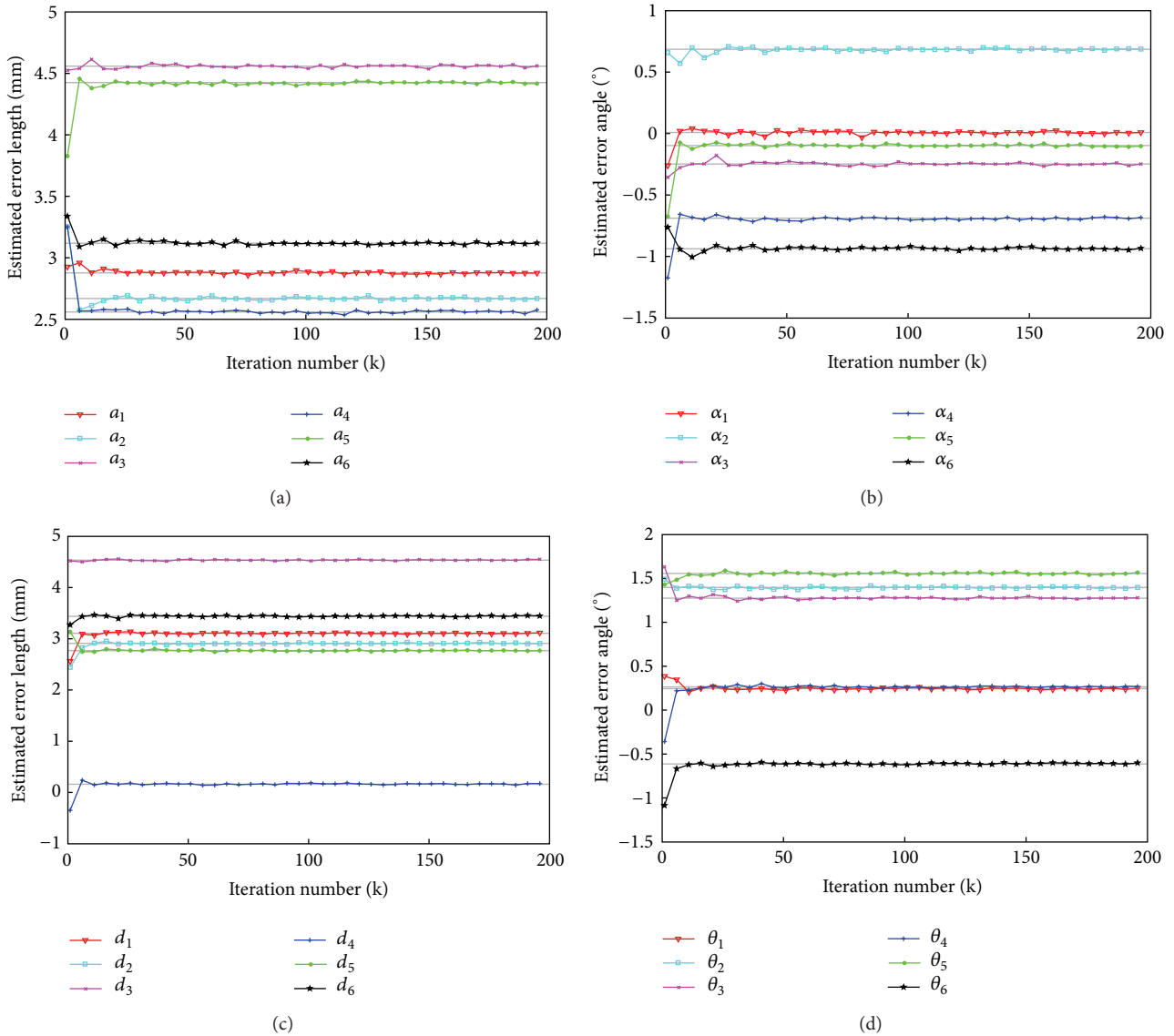


FIGURE 7: Estimated *D-H* parameter errors with EKF.

- (2) Manipulator orientation estimation: the orientations of the manipulator were estimated via the KFs from the obtained data.
- (3) Kinematic parameters identification: the manipulator kinematic parameters were identified from the estimated orientations and the joint data.
- (4) Calibration accuracy assessment: 3D pose errors were used to verify the calibration accuracy by inserting a peg into the holes (Figure 5).

Table 1 lists the nominal robot link parameters of the robot, which were chosen as the initial conditions for the above kinematic identification algorithm, and Figure 4 shows the skeleton of the GOOGOL GRB3016 robot with all coordinate frames and geometric features. As expected, the increase in the noise intensity will lead to the increase in the calibration errors.

From Table 1 note that a GOOGOL GRB3016 robot with 6 DOF needs 24 geometric parameters to be modeled. From

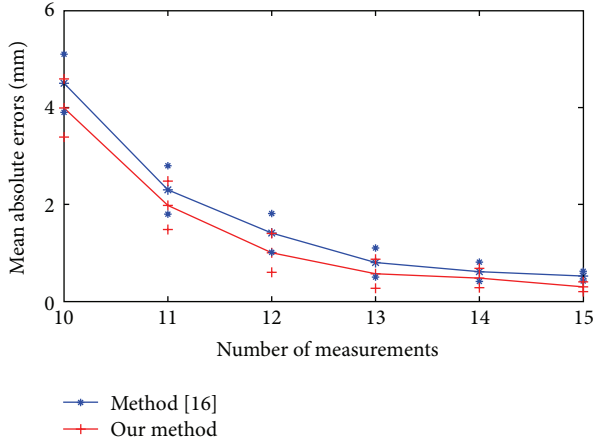


FIGURE 8: Mean absolute errors in different number of measurements.

(18), each 3D robot orientation provides 3 model equations. So a unique computation of the 24 parameters needs 8 pose measurements at least. And more pose measurements will decrease the calibration errors. But limited by the measurement accuracy (affected by the noise), the calibration errors intend to be stable as the pose measurements increase.

In order to validate the proposed IMU-based robot calibration, we have performed a number of peg-into-hole experiments. In our experiments, two calibration methods were used to carry out the experiments of peg-into-hole. Our method was initially compared with a standard vision-based robot calibration of the Meng and Zhuang [16]. There were 16 holes in the steel plate (Figure 5) and 16 tests of peg-into-hole were carried out in each experiment. The peg was the cylinder with 7.5 mm in radius and 150 mm in length. The radius of the hole was 8 mm. The size of the steel plate was 300 mm × 500 mm. In the step of our method, an IMU (Xsens MTi-100 IMU) was rigidly attached to the robot tool flange to measure the pose of the robot tool. The static accuracy of roll and pitch was 0.02 deg and that of yaw was 0.05 deg. The dynamic accuracy of roll and pitch was 0.05 deg and that of yaw was 0.1 deg. The noise density of gyroscopes was 0.01 deg/s/√Hz and that of the magnetometer was 200 μG √Hz. The IMU measurements were received at 100 Hz. Since the robot will be stopped after it executes a command, in order to improve the accuracy, the system collects the static IMU data after the robot stops. The coordinates of the center of the hole with respect to the base coordinate system were known. In the step of method [16], a camera, with 1280(H) × 960(V) picture elements and 22 frame/s frequency, was mounted on the robot tool to capture the RGB images of the chessboard. A chessboard pattern, the position of which was unknown in the reference frame, was placed on the platform. The distances, measured by a caliper, between two adjacent corners of the square were known. Note that the robot has different kinematic parameter errors in the different position. In order to improve the accuracy, the system made a robot calibration before inserting the peg into each hole.

TABLE 1: The nominal link parameters in DH model for the GOOGOL GRB3016 robot.

Joints	DH			
	a	α	d	θ
1	150	$-\pi/2$	250	0
2	570	$-\pi$	0	$-\pi/2$
3	150	$\pi/2$	0	0
4	0	$-\pi/2$	650	0
5	0	$-\pi/2$	0	$-\pi/2$
6	0	0	-200	0

To evaluate the calibration accuracy, 3D position errors between the peg and the hole were proposed. A calibrated camera with 1280(H) × 960(V) picture elements was used to measure the 3D position errors (Figure 5). In the initiation of the system, the camera measured the coordinates of the center of the hole with respect to the camera coordinate system. After the peg was inserted into a hole, the camera can measure the depth and the direction of insertion by detecting the edge of the peg. And then the system could calculate the center of the peg with respect to the camera coordinate system. Let (x_o, y_o, z_o) and (x_t, y_t, z_t) be the centers of the peg and the hole, respectively (Figure 6). Then the 3D position error E can be written:

$$E = \sqrt{(x_o - x_t)^2 + (y_o - y_t)^2 + (z_o - z_t)^2}. \quad (28)$$

The mean absolute error in position for N peg-into-hole tests was

$$E_m = \sum_{k=1}^N \frac{E_k}{N}, \quad (29)$$

where E_k is the 3D position error for the k th hole.

6.2. Result Analysis. Following the initialization step, the system performs a series of tests for peg-into-hole. The KF processes the IMU measurements and concurrently estimates the state vector. For EKF, the number of maximum iteration was set as 2000. Q and R were set as $1.0 \times 10^{-3} \times I_{3 \times 3}$. When the state vector was less than 1.0×10^{-5} , the iteration terminated. The proposed algorithm was evaluated by estimating the orientation from the IMU and the theoretical orientation obtained by using (3). Figure 7 shows the estimated DH parameter error values of the robot tool. As shown in Figure 7, the EKF algorithm quickly converged to the stable error value from about the fifteenth iteration.

Table 2 shows the estimated parameter errors when the calibration sets were used in EKF fro 200 iterations. Since the EKF algorithm quickly converged to the stable error value from about the fifteenth iteration, the iteration should be set as 15 so that the calibration could be carried out in real time.

In the experiments of peg-into-hole, the iteration was set as 15 to estimate the parameter errors. The state-estimate errors with 15 pose measurements are shown in Table 3. As evident from Table 3, even with the noise error for the IMU measurements, the algorithm is still able to attain

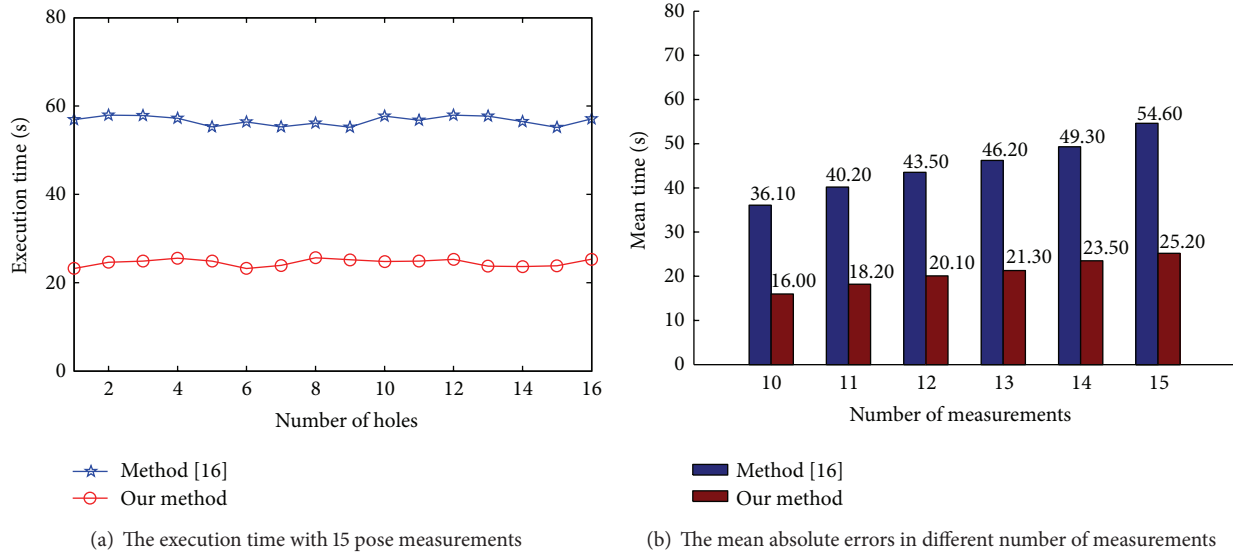


FIGURE 9: The comparison in execution time.

TABLE 2: Estimated parameter errors of 6 DOF robot.

Error	Joint					
	1	2	3	4	5	6
Δa (mm)	2.8759	2.6679	4.5582	2.5580	4.4248	3.1172
$\Delta \alpha$ (°)	0.0088	0.6855	-0.2482	-0.6892	-0.0974	-0.9373
Δd (mm)	3.1049	2.9095	4.5347	0.1594	2.7675	3.4382
$\Delta \theta$ (°)	0.2441	1.3982	1.2752	0.2644	1.5575	-0.6128

TABLE 3: The 3D errors with 15 pose measurements.

Hole number	Method [16] (mm)	Our method (mm)
1	0.45	0.40
2	0.33	0.30
3	0.51	0.29
4	0.52	0.21
5	0.38	0.32
6	0.61	0.16
7	0.61	0.37
8	0.49	0.46
9	0.53	0.23
10	0.51	0.38
11	0.48	0.42
12	0.57	0.14
13	0.44	0.15
14	0.71	0.35
15	0.48	0.26
16	0.51	0.36

very accurate estimates of the calibration parameters. Table 3 shows the accuracy result of method [16] too. By comparing the results of Table 3, the calibration parameters were more accurate in our method.

In our method, a unique computation of the 24 kinematic parameters needs 8 pose measurements. In method [16], it needs 4 pose measurements at least. We compare two methods from 10 pose measurements to 15 pose measurements. The results presented in Figure 8 show that with more pose measurements, the parameter error decreases gradually. The mean absolute estimation errors from 10 pose measurements to 15 pose measurements in method [16] were 4.5 mm, 2.3 mm, 1.41 mm, 1.3 mm, 0.61 mm, and 0.52 mm with standard deviations (SDs) of 0.4 mm, 0.38 mm, 0.35 mm, 0.28 mm, 0.09 mm, and 0.09 mm. Compared with method [16], the mean absolute errors of our method drop about 0.51 mm, 0.32 mm, 0.41 mm, 0.23 mm, 0.13 mm, and 0.22 mm. The estimation errors trend slows down after two times of the minimum number of poses measurements.

Compared with method [16], the great advantage of our method is that the system does not need to make a more motion to take the photo. After the robot executed a command, the robot would stop and the system concurrently obtained the static measurement data from the IMU. Figure 9(a) shows the execution time of two methods with 15 pose measurements. In method [16], the average time of taking a photo was 3 s. In our method, the station time of robot was about 1 s. The execution time of peg-into-hole is about 8 s. The time of parameter identification was 0.8 s. Figure 9(b) shows the comparison of execution time with different pose measurements. The time in method [16] is more than two times that in our method.

7. Conclusions

An IMU-based online autonomous calibration for serial robot has been proposed in this paper. In this approach, the IMU is rigidly attached to the robot tool to estimate the robot pose automatically during the working time. An efficient approach which incorporates Factored Quaternion Algorithm (FQA), Kalman Filter (KF), and Extended Kalman Filter (EKF) to estimate the orientation of the IMU is presented in this paper. After the robot poses are estimated, the kinematics identification can be carried out. Finally, the robot kinematic parameters can be corrected from the identification results in real time. The whole procedure of the robot calibration is automatic and without any manual intervention. The results of the experiments show the good accuracy, convenience, and effectiveness of the presented approach.

Compared with the existing expensive and complex approach, the method this paper proposed is easier to use and setup. Compared to the existing vision-based self-calibration method, the proposed method can conduct the calibration more accurately and with less execution time. In the future work, we will research the approach which can accurately estimate the robot pose without stopping the robot. With the dynamic pose measurements, the robot calibration will be more efficient.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] M. Vincze, J. P. Prenninger, and H. Gander, "Laser tracking system to measure position and orientation of robot end effectors under motion," *International Journal of Robotics Research*, vol. 13, no. 4, pp. 305–314, 1994.
- [2] H. Zhuang, J. Yan, and O. Masory, "Calibration of Stewart platforms and other parallel manipulators by minimizing inverse kinematic residuals," *Journal of Robotic Systems*, vol. 15, no. 7, pp. 395–405, 1998.
- [3] H. Zhuang, "Self-calibration of parallel mechanisms with a case study on Stewart platforms," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 3, pp. 387–397, 1997.
- [4] W. Khalil and S. Besnard, "Self calibration of Stewart-Gough parallel robots without extra sensors," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 6, pp. 1116–1121, 1999.
- [5] I.-W. Park, B.-J. Lee, S.-H. Cho, Y.-D. Hong, and J.-H. Kim, "Laser-based kinematic calibration of robot manipulator using differential kinematics," *IEEE/ASME Transactions on Mechatronics*, vol. 17, no. 6, pp. 1059–1067, 2012.
- [6] D. J. Bennett and J. M. Hollerbach, "Autonomous calibration of single-loop closed kinematic chains formed by manipulators with passive endpoint constraints," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 5, pp. 597–606, 1991.
- [7] D. J. Bennett, D. Geiger, and J. M. Hollerbach, "Autonomous robot calibration for hand-eye coordination," *International Journal of Robotics Research*, vol. 10, no. 5, pp. 550–559, 1991.
- [8] S. K. Mustafa, G. Yang, S. H. Yeo, W. Lin, and I.-M. Chen, "Self-calibration of a biologically inspired 7 DOF cable-driven robotic arm," *IEEE/ASME Transactions on Mechatronics*, vol. 13, no. 1, pp. 66–75, 2008.
- [9] H. Zhuang, Z. S. Roth, and K. Wang, "Robot calibration by mobile camera systems," *Journal of Robotic Systems*, vol. 11, no. 3, pp. 155–167, 1994.
- [10] P. Renaud, N. Andreff, F. Marquet, and P. Martinet, "Vision-based kinematic calibration of a H4 parallel mechanism," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '03)*, pp. 1191–1196, September 2003.
- [11] D. Daney, N. Andreff, and Y. Papegay, *Interval Method for Calibration of Parallel Robots: A Vision-Based Experimentation*, Computational Kinematics, Cassino, Italy, 2005.
- [12] P. Renaud, N. Andreff, J.-M. Lavest, and M. Dhome, "Simplifying the kinematic calibration of parallel mechanisms using vision-based metrology," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 12–22, 2006.
- [13] P. Renaud, N. Andreff, M. Philippe, and G. Gogu, "Kinematic calibration of parallel mechanisms: a novel approach using legs observation," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 529–538, 2005.
- [14] N. Andreff and P. Martinet, "Unifying kinematic modeling, identification, and control of a Gough-Stewart parallel robot into a vision-based framework," *IEEE Transactions on Robotics*, vol. 22, no. 6, pp. 1077–1086, 2006.
- [15] N. Andreff, P. Renaud, P. Martinet, and F. Pierrot, "Vision-based kinematic calibration of an H4 parallel mechanism: practical accuracies," *Industrial Robot*, vol. 31, no. 3, pp. 273–283, 2004.
- [16] Y. Meng and H. Zhuang, "Autonomous robot calibration using vision technology," *Robotics and Computer-Integrated Manufacturing*, vol. 23, no. 4, pp. 436–446, 2007.
- [17] J. M. S. T. Motta, G. C. De Carvalho, and R. S. McMaster, "Robot calibration using a 3D vision-based measurement system with a single camera," *Robotics and Computer-Integrated Manufacturing*, vol. 17, no. 6, pp. 487–497, 2001.
- [18] Y. Zhu, C. Hu, J. Hu, and K. Yang, "Accuracy and simplicity oriented self-calibration approach for two-dimensional precision stages," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 6, pp. 2264–2272, 2012.
- [19] C. S. Gatla, R. Lumia, J. Wood, and G. Starr, "An automated method to calibrate industrial robots using a virtual closed kinematic chain," *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1105–1116, 2007.
- [20] F. Aghili, "Adaptive control of manipulators forming closed kinematic chain with inaccurate kinematic model," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 5, pp. 1544–1554, 2012.
- [21] G. Alici and B. Shirinzadeh, "Enhanced stiffness modeling, identification and characterization for robot manipulators," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 554–564, 2005.
- [22] W. S. Newman, C. E. Birkhimer, R. J. Horning, and A. T. Wilkey, "Calibration of a Motoman P8 robot based on laser tracking," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '00)*, pp. 3597–3602, San Francisco, Calif, USA, April 2000.
- [23] A. Omodei, G. Legnani, and R. Adamini, "Three methodologies for the calibration of industrial manipulators: experimental results on a SCARARobot," *Journal of Robotic Systems*, vol. 17, no. 6, pp. 291–307, 2000.
- [24] A. Omodei, G. Legnani, and R. Adamini, "Calibration of a measuring robot: experimental results on a 5 DOF structure," *Journal of Robotic Systems*, vol. 18, no. 5, pp. 237–250, 2001.

- [25] G. Alici and B. Shirinzadeh, "A systematic technique to estimate positioning errors for robot accuracy improvement using laser interferometry based sensing," *Mechanism and Machine Theory*, vol. 40, no. 8, pp. 879–906, 2005.
- [26] G. Alici, R. Jagielski, Y. Şekercioğlu, and B. Shirinzadeh, "Prediction of geometric errors of robot manipulators with Particle Swarm Optimisation method," *Robotics and Autonomous Systems*, vol. 54, no. 12, pp. 956–966, 2006.
- [27] J. O'Brian, R. Bodenheimer, G. Brostow, and J. Hodgins, "Automatic joint parameter estimation from magnetic motion capture data," in *Proceedings of the Graph Interface*, pp. 53–60, May 2000.
- [28] P. Renaud, N. Andreff, J.-M. Lavest, and M. Dhome, "Simplifying the kinematic calibration of parallel mechanisms using vision-based metrology," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 12–22, 2006.
- [29] A. Rauf, A. Pervez, and J. Ryu, "Experimental results on kinematic calibration of parallel manipulators using a partial pose measurement device," *IEEE Transactions on Robotics*, vol. 22, no. 2, pp. 379–384, 2006.
- [30] J. Santolaria, J.-J. Aguilar, J.-A. Yagüe, and J. Pastor, "Kinematic parameter estimation technique for calibration and repeatability improvement of articulated arm coordinate measuring machines," *Precision Engineering*, vol. 32, no. 4, pp. 251–268, 2008.
- [31] R. He, Y. Zhao, S. Yang, and S. Yang, "Kinematic-parameter identification for serial-robot calibration based on POE formula," *IEEE Transactions on Robotics*, vol. 26, no. 3, pp. 411–423, 2010.
- [32] G. Gatti and G. Danieli, "A practical approach to compensate for geometric errors in measuring arms: application to a six-degree-of-freedom kinematic structure," *Measurement Science and Technology*, vol. 19, no. 1, Article ID 015107, 2008.
- [33] K. Schröer, S. L. Albright, and M. Grethlein, "Complete, minimal and model-continuous kinematic models for robot calibration," *Robotics and Computer-Integrated Manufacturing*, vol. 13, no. 1, pp. 73–85, 1997.
- [34] H. Zhuang and Z. Roth, *Camera -Aided Robot Calibration*, CRC, Boca Raton, Fla, USA, 1996.
- [35] S. A. Hayati, "Robot arm geometric link parameter estimation," in *Proceedings of the 22th IEEE Conference on Decision and Control*, vol. 22, pp. 1477–1483, December 1983.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

