

# Online pose correction of an industrial robot using an optical coordinate measure machine system

Sepehr Gharaaty<sup>1</sup>, Tingting Shu<sup>1</sup> , Ahmed Joubair<sup>2</sup>,  
Wen Fang Xie<sup>1</sup> and Ilian A Bonev<sup>2</sup>

## Abstract

In this article, a dynamic pose correction scheme is proposed to enhance the pose accuracy of industrial robots. The dynamic pose correction scheme uses the dynamic pose measurements as feedback to accurately guide the robot end-effector to the desired pose. The pose is measured online with an optical coordinate measure machine, that is, C-Track 780 from Creaform. A root mean square method is proposed to filter the noise from the pose measurements. The dynamic pose correction scheme adopts proportional-integral-derivative controller and generates commands to the FANUC robot controller. The developed dynamic pose correction scheme has been tested on two industrial robots, FANUC LR Mate 200iC and FANUC M20iA. The experimental results on both robots demonstrate that the robots can reach the desired pose with an accuracy of  $\pm 0.050$  mm for position and  $\pm 0.050^\circ$  for orientation. As a result, the developed pose correction can make the industrial robots meet higher accuracy requirement in the applications such as riveting, drilling, and spot welding.

## Keywords

Pose correction, visual servoing, pose accuracy, optical CMM, accuracy enhancement

Date received: 26 February 2018; accepted: 18 June 2018

Topic: Robot Manipulation and Control

Topic Editor: Andrey V Savkin

Associate Editor: Yongping Pan

## Introduction

Industrial robots have revolutionized various industries. Welding, drilling, assembling, and riveting are some of the frequent applications of industrial robots in aerospace, manufacturing, and automation industries. For instance, in aircraft assembly, industrial robots are widely used due to their high precision and adaptability.<sup>1</sup> However, the low absolute accuracy of industrial robots has negatively impacted their high potential, especially in the aerospace industry, where the maximum position error tolerated is typically in the range of  $\pm 0.1$  mm. Sources of pose (i.e. position and orientation) errors in an industrial robot can vary from hardware and software limitations, to the manufacturing tolerances of the mechanical components, and to the environment where the robot is being operated.<sup>2</sup> These

sources of errors can be attributed to the following aspects: geometric, dynamic, thermal, and system.<sup>3</sup>

In recent years, improving the accuracy of industrial robots has been a major concern for researchers, manufacturers, and customers in some industries. Robot calibration is a common way to enhance the positioning accuracy of

<sup>1</sup> Faculty of Engineering and Computer Science, Department of Mechanical, Industrial and Aerospace Engineering (MIAE), Concordia University, Montreal, Quebec, Canada

<sup>2</sup> Superior Technology School, Montreal, Quebec, Canada

### Corresponding author:

Tingting Shu, Concordia University, 1455 De Maisonneuve W, Montreal, Quebec H3G 1M8, Canada.

Email: anny.buaa@gmail.com



Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License

(<http://www.creativecommons.org/licenses/by/4.0/>) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (<https://us.sagepub.com/en-us/nam/open-access-at-sage>).

industrial robots.<sup>4,5</sup> There are two main categories of robot calibration: kinematic and non-kinematic approaches. Kinematic calibration deals only with the kinematic parameters of the robot. In contrast, non-kinematic calibration deals with other sources of error as well.<sup>5,6</sup> In terms of the procedures, robot calibration consists of four major steps: modeling, measurements, parameter identification, and accuracy validation. Motta et al. performed a kinematic off-line robot calibration on an ABB (Saint-Laurent, Quebec) IRB 2400 robot and a PUMA (Danbury, Connecticut) 500 robot in different workspace volumes using a three-dimensional (3-D) vision-based measurement system.<sup>7</sup> According to their proposed system, the median error for the ABB IRB 2400 robot was reduced from 1.75 mm to 0.60 mm, while the error for PUMA 500 was reduced from 2.03 mm to 0.33 mm. Song et al. evaluated and improved the positioning accuracy of a five-degree-of-freedom (5-DOF) medical robotic system through off-line kinematic calibration.<sup>8</sup> The accuracy was improved to 0.888 mm and 1.142 mm for the mean and maximum values, respectively. Joubair et al. performed the kinematic and non-kinematic calibration on a FANUC (Mississauga, Ontario) LR Mate 200iC using a laser tracker for position measurements.<sup>9</sup> For the non-kinematic calibration, they considered five additional stiffness parameters. When considering only the kinematic parameters, their proposed calibration process improved the robot's mean position error from 0.622 mm to 0.250 mm and the maximum position error from 1.762 mm to 1.062 mm. However, the non-kinematic calibration reduced the mean position error to 0.142 mm and the maximum position error to 0.899 mm.

In 2014, Nubiola et al. used a C-Track (an optical coordinate measure machine (CMM)) for the first time in the calibration process of an ABB IRB 120.<sup>10</sup> In the calibration model, five stiffness parameters were used in addition to the kinematic parameters. After calibration, the robot's mean position error was improved from 3 mm to 0.150 mm, and the maximum position error was improved from 5 mm to 0.500 mm. More recently, Gaudreault et al. used a new, low-cost 3-D measuring device (TriCal, Quebec, Canada) to enhance the absolute accuracy of an ABB IRB 120 robot.<sup>11</sup> In their experiments, TriCal is used in a target workspace to record the position measurements for calibration process. Using this device and the corresponding calibration procedure, the robot's mean position error was reduced from 1.114 mm to 0.272 mm, and the maximum position error was reduced from 1.726 mm to 0.491 mm. In addition to the above mentioned categories for robot calibration, the analysis of industrial robots' accuracy is divided into two categories: absolute accuracy assessment and relative accuracy (i.e. Euclidean distance accuracy) assessment. For example, Joubair et al. implemented a kinematic calibration on a FANUC LR Mate 200iC robot without external measurement sensors.<sup>12</sup> Instead, they used a touch probe mounted on the robot and an artefact composed of precision balls and improved accuracy of the robot. The robot's mean position

error was reduced from 0.698 mm to 0.086 mm and the maximum position error from 1.321 mm to 0.127 mm.

Undoubtedly, robot calibration yields satisfactory results for a multitude of applications, where position error of up to 0.5 mm are acceptable. Further improving the accuracy of an industrial robot is theoretically possible but still impossible to achieve in practice through calibration. The effects that are very difficult to model, such as thermal expansion, can easily account for 0.2 mm of position error. Furthermore, even if a robot is calibrated, users often fail at accurately measuring/identifying their tools and robot cells. It is possible to perform a full robot cell calibration on site, but this is very expensive. However, because of wear and tear and other possible impacts, a robot must be calibrated frequently. Robot calibration is therefore not only limited in its efficiency but also expensive in the long term.

As long as conditions permit (e.g. space and visibility), visual servoing is a much better alternative.<sup>13,14</sup> Not only may it allow robots to be as accurate as the measurement system itself, but it also improves the pose accuracy of a robot. With the development of high-precision measurement sensors, visual servoing has been deployed to correct the position and orientation of the robot end-effector in real time. Bone and Capson developed a vision-guided robotic assembly work cell.<sup>15</sup> In their work, a 2-D computer vision was used to correct the initial placement of the parts, and a 3-D computer vision was used to correct the pose of the parts before joining them together. Lertpiriyasuwat and Berg used a laser tracker and Kalman filter to estimate the position and orientation of the end-effector of a 6-DOF gantry robot in real time and compensate the pose errors dynamically.<sup>16</sup> Norman et al. used an indoor GPS technology as a feedback measurement sensor and improved the absolute accuracy of a KUKA (Mississauga, Ontario) robot in real time.<sup>17</sup> Also, Jin et al. proposed a robot-assisted assembly system to improve the accuracy in the process of installing small components.<sup>18</sup> In this system, a laser tracker was integrated with a KUKA KR360L280-2 robot in a closed-loop system, and the maximum position error was reduced from 1.379 mm to 0.069 mm, after two or three correction iterations. However, this robot was calibrated by the manufacturer before the experiment. In addition, the authors did not specify the required time to achieve the mentioned accuracy. Recently, Posada et al. proposed and implemented an external sensor-based compensation method to identify and reduce the errors in a robotic drilling process to 0.1 mm for position and 0.2° for orientation.<sup>19</sup> They also used a laser tracker as a feedback measurement sensor and corrected the pose of a KUKA KR210 robot online. In their work, it was proved that after five iterations, which took approximately 125 s, the threshold for the precision before the drilling (0.1 mm and 0.2°) was satisfied. After drilling, the average Euclidean distance error was 0.285 mm.

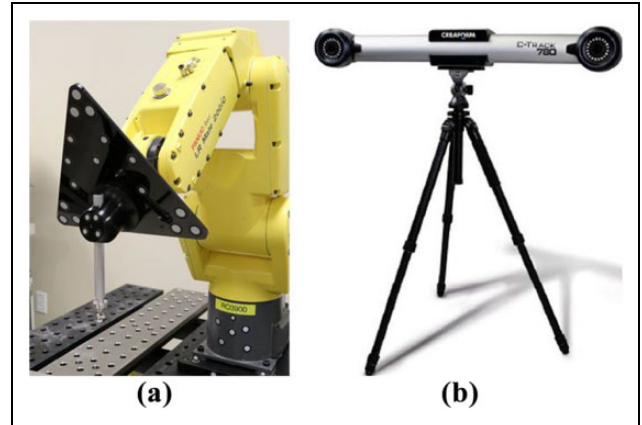
However, laser trackers are very expensive, especially if equipped with options for orientation measurements in addition to position measurements. Furthermore, laser

trackers can measure the position and the possible orientation of only one body at a time. As it is impossible for the laser tracker to be solidly attached to the workpiece, vibrations and air currents on the factory floor can significantly lower the measurement precision. Therefore, in practice, optical CMMs provide a more cost-effective alternative, even as they are less accurate in a laboratory setting.

Some research work has been dedicated to using variants of the proportional-integral-derivative (PID) controls to improve the performance of mechatronic systems.<sup>20,21</sup> A few researchers have adopted CMM for pose correction of industrial robots.<sup>22,23</sup> For example, an off-line compensation method is presented in the study by Nikon,<sup>22</sup> which measures the sources of robot error and applies them to predict reference pose off-line using Nikon (Mississauga, Ontario) K-Series optical CMM. Although it is reported that the robot accuracy can be increased to 0.2 mm, this method needs experts to carry out extensive tests to predict the reference pose. In the study by Hsiao and Huang,<sup>23</sup> an iterative learning control combined with a PD feedback control is applied to improve the tracking performance of an industrial robot. The accuracy is demonstrated in joint space and maximum error of the joint angle is larger than  $0.1^\circ$ . In addition, the pose accuracy is not equivalent to the joint angle accuracy, which is based on the kinematic model with both the geometric and non-geometric errors. Hence, an online compensation method is demanded, which does not need off-line reference pose prediction and can directly correct the pose using the measurements from CMM in real time.

In this article, a dual-camera optical CMM (C-Track 780 from Creaform [Quebec, Canada]) is used as a measurement sensor for the single-pose correction of a 6-DOF FANUC industrial robot. Based on the measured pose as feedback, a novel online pose correction algorithm is developed and implemented on two FANUC industrial robots (FANUC LR Mate 200iC and FANUC Mi20A). The implementation of the developed pose correction scheme can be performed in two options—with FANUC's dynamic path modification (DPM) module and without the DPM module. A comparison between two implementations are carried out and the controllers with both implementations can achieve an accuracy of  $\pm 0.050$  mm for position and  $\pm 0.050^\circ$  for orientation.

This article is organized as follows. First, a general description of the experimental setup is presented. Then, the repeatability of the robot used in this work is measured using the TriCal, and a kinematic calibration is performed using the C-Track.<sup>11</sup> Next, C-Track data analysis is given. The static calibration is carried out for comparison purposes. The obtained accuracy after calibration is compared with the results on two implementations of the developed closed-loop pose correction algorithm. The detailed implementations are explained, and then the experimental results and verification methods are presented. Finally, the conclusion and future works are discussed.



**Figure 1.** Experimental equipment: (a) FANUC LR Mate 200iC (b) C-Track from Creaform.

## System description

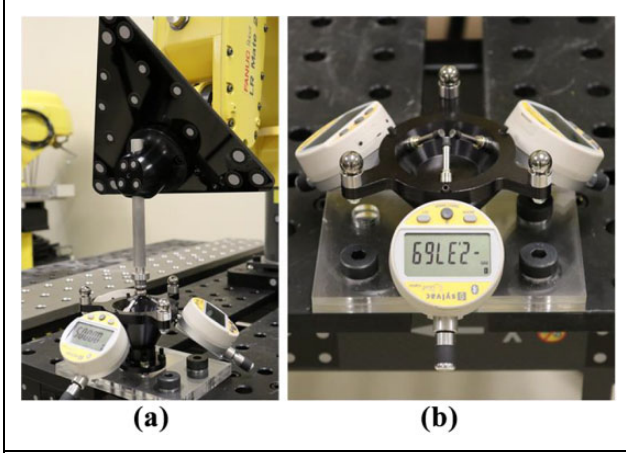
The goal of the research is to improve the pose accuracy of the robot at a single pose (e.g. for applications such as drilling or riveting). However, the compensation is done dynamically, using a closed-loop pose correction system. The precision considered for this research is  $\pm 0.050$  mm for position and  $\pm 0.050^\circ$  for orientation. The experimental setup is composed of an industrial robot, a C-Track for online pose measurements, and a custom-made 3-D measurement device for verification.

The industrial robots used in this research are two 6-DOF serial robots, FANUC LR Mate 200iC and FANUC Mi20A. FANUC LR Mate 200iC is a 6-DOF robot with a payload of 5 kg, while FANUC Mi20A is a 6-DOF robot with a payload of 20 kg. The C-Track is a binocular vision sensor that can measure the position of a number of retroreflective targets (often retroreflective stickers) simultaneously at the maximum rate of 29 Hz. In this research, the C-Track is used to measure the pose of the robot end-effector with respect to (w.r.t.) a workpiece. To evaluate the position and orientation of the robot, at least four retroreflectors are required. Therefore, a tool is designed in a way that several targets can be attached. The FANUC LR Mate 200iC robot with the designed tool and C-Track is shown in Figure 1.

## Repeatability determination

The repeatability of the robot is defined as the ability of the robot to move back to the same position and orientation in its workspace.<sup>24</sup> In the process of repeatability evaluation, we used a custom-made 3-D position measuring device to measure the position error of the robot at a given pose. The device consists of the three orthogonally mounted digital indicators as shown in Figure 2.

In order to determine its repeatability, the robot is programmed to move its end-effector to the predefined coordinates recorded in the robot controller. When the robot



**Figure 2.** Experimental setup for repeatability test: (a) robot tool (b) TriCal.

reaches the desired pose for the first time, the indicators are set to zero. Then the robot is programmed to move to the same pose several times, where the offset measured by the indicators is recorded at each iteration. The experimental setup for this test is shown in Figure 2. According to ISO-9283 standard, the repeatability of the robot can be calculated using 30 iterations and the following equations

$$L_i = \sqrt{E_x^2 + E_y^2 + E_z^2} \quad (1)$$

$$\bar{L} = \frac{1}{N} \sum_{i=1}^N L_i \quad (2)$$

$$S_L = \sqrt{\frac{\sum_{i=1}^N (L_i - \bar{L})^2}{N - 1}} \quad (3)$$

$$RP = \bar{L} + 3 \times S_L \quad (4)$$

where  $N$  is the number of iterations,  $E_x$ ,  $E_y$ , and  $E_z$  are the measurements of the indicators for each iteration, and  $RP$  is the obtained repeatability of the robot.

According to the above equations and the experimental results for 30 iterations on FANUC LR Mate 200iC, the calculated repeatability of the robot is 0.021 mm. The repeatability of FANUC Mi20A is 0.08 mm. We should mention that the repeatability evaluation was required in order to confirm that its value is still acceptable (i.e. significantly better than the expected accuracy, in our calibration process), since the robots are no longer brand new. The test is repeated at several other positions and similar results are obtained.

## C-Track data analysis

One of the main limitations of the C-Track is the noise associated with its measurements, which can affect the pose estimation accuracy. In order to filter the measurement noises, the root mean square (RMS) technique is used.

Experimental analyses prove that using RMS value of 10 measurements from a stationary reference model can improve the measurements up to 75%.<sup>25</sup> It is noted that a reference model represents a reference frame associated with a set of targets measured by the C-Track. Figure 3 shows 100 measurements taken by the C-Track of the robot end-effector position in a fixed configuration where the vibrations of the robot and the C-Track are negligible.

As can be seen in Figure 3, the peak to peak difference for  $X$ ,  $Y$ , and  $Z$  coordinates are approximately 0.060 mm, 0.050 mm, and 0.120 mm, respectively. Therefore, achieving an accuracy better than 0.120 mm would be impossible with the current measurements. In order to address this issue, RMS is used to filter the noise of the C-Track measurements. The RMS value of the pose of a model measured by C-Track can be obtained from the following equation

$$\vec{V}_{RMS} = \text{sgn}(\vec{V}_i) \sqrt{\frac{\sum_{i=1}^N \vec{V}_i^2}{N}} \quad (5)$$

where  $\vec{V}_i$  is a vector of six components including position and orientation of the robot end-effector model measured by the C-Track at each interval,  $N$  is the number of measurements to calculate RMS, that is, RMS length, and  $\vec{V}_{RMS}$  is the vector of the calculated pose based on the RMS value of  $N$  measurements. After analyzing the measured data, it is proved that the difference between the RMS values of each 10 measurements is almost negligible. Figure 4 shows the RMS values for the same 100 measurements.

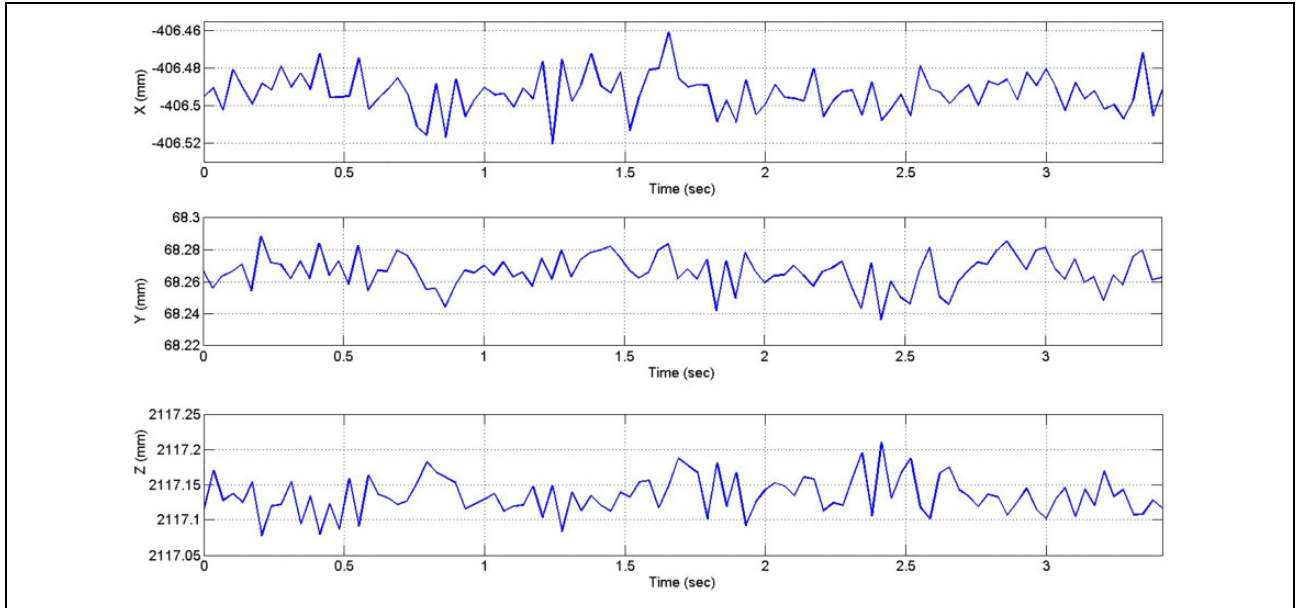
As shown in the Figure 4, the maximum deviation of RMS values on  $X$ ,  $Y$ , and  $Z$  axes is reduced to 0.015 mm, 0.020 mm, and 0.030 mm, respectively. Hence, using the RMS value, the noise of the C-Track measurements can be improved by 75%. This solution provides sufficient accuracy for the measured data and can be used as the feedback signal in the dynamic pose correction (DPC) scheme.

## Accuracy enhancement

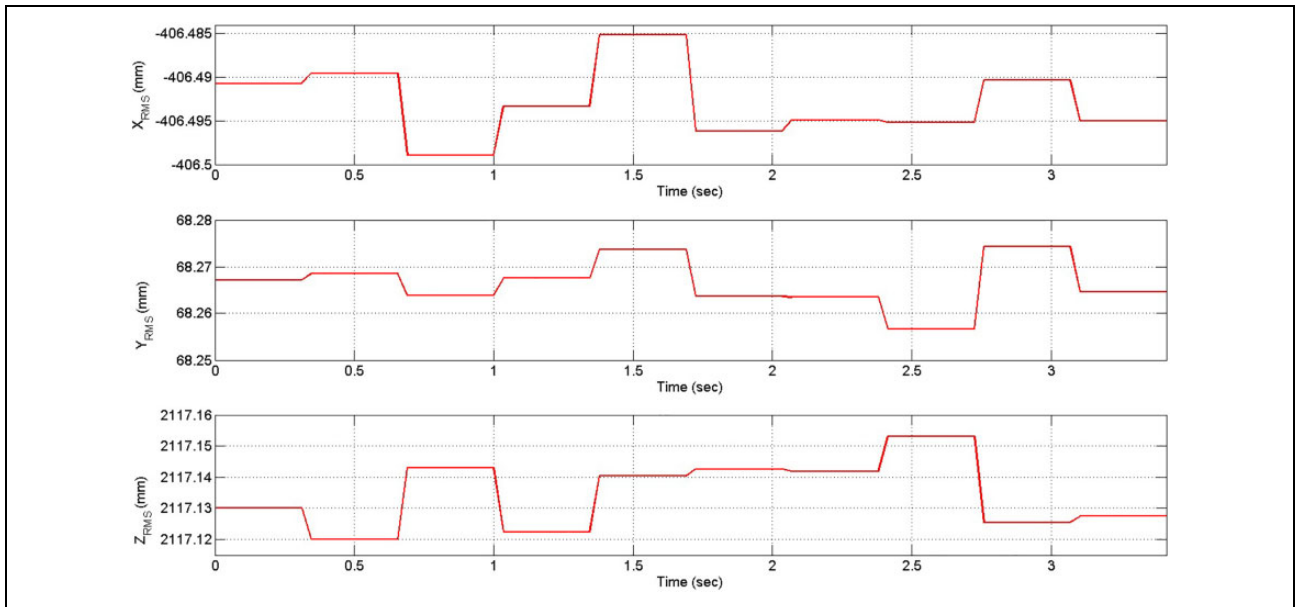
### Static calibration

In order to have an objective comparison, a static calibration is performed using the C-Track on the robots. For this purpose, the position of the robot end-effector (FANUC LR Mate 200iC, for example) was recorded by the C-Track in 800 arbitrary configurations (i.e. sets of joint angles), and at the same time, the joint angles for each configuration were collected from the robot controller. The whole procedure was fully automated using an interface developed in MS Visual Basic [version 2013].

Among the 800 configurations, 100 configurations were selected through an observability analysis (Figure 5) and used for the parameter identification, while the remaining 700 were used for validation. The identification process



**Figure 3.** Hundred measurements taken by C-Track from the position of a fixed model.



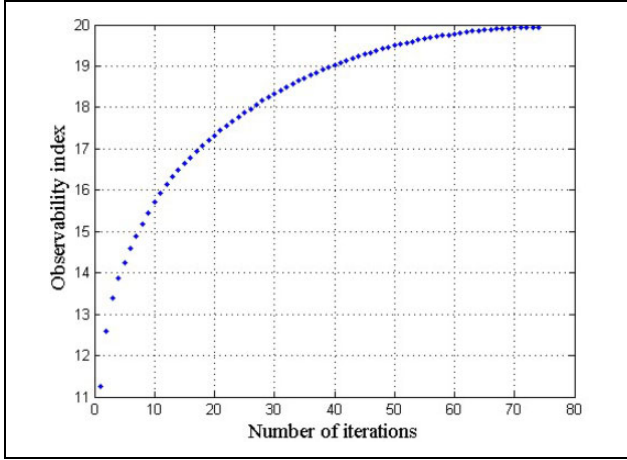
**Figure 4.** RMS values for each 10 measurements. RMS: root mean square.

was based on minimizing the position residuals of the robot end-effector. The experimental results are shown in Figures 6 and 7.

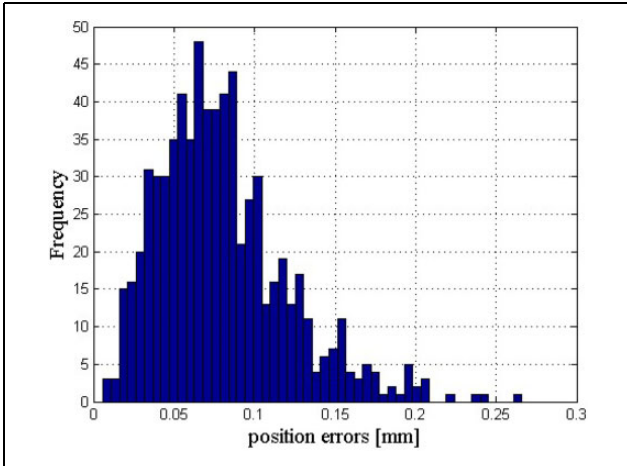
In this experiment, the position of the robot end-effector is obtained using the RMS value of 80 measurements with the C-Track at each robot configuration (the 80 measurements take less than 3 s). The calibration results show that the robot mean position error has been improved to 0.092 mm, and the maximum position error has been improved to 0.266 mm.

### *Dynamic closed-loop pose correction*

In this section, the pose error calculation between the desired pose  $\vec{v}_{Des}$  and the current pose  $\vec{v}_{Cur}$  of the end-effector is presented. Then, the control law is designed to improve the robot's pose accuracy. The DPC algorithm is proposed to reduce the pose error. The developed DPC algorithm is implemented with two options: with DPM and without DPM in an independent computer. The block diagram of the DPC control scheme is shown in Figure 8.



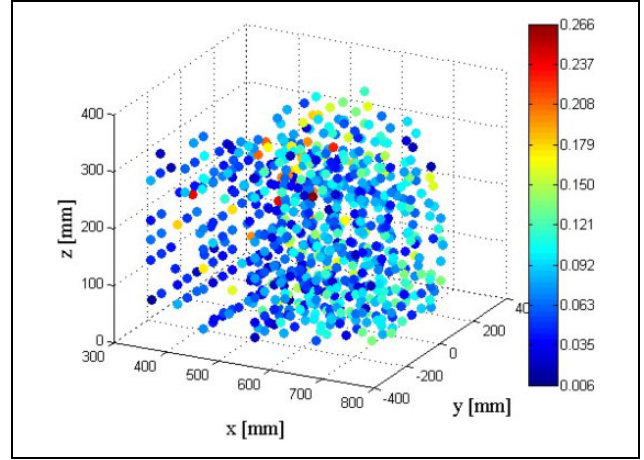
**Figure 5.** Convergence of the observability index  $O_1$ .



**Figure 6.** Histogram of the position accuracy after calibration as measured in 700 robot configurations.

The schematic of the experimental setup is shown in Figure 9. There are four coordinate systems used in this experiment: tool frame  $\{T\}$ , user frame  $\{U\}$ , workpiece frame  $\{W\}$ , and C-Track frame  $\{C\}$ . The coordinates of frames  $\{T\}$ ,  $\{U\}$ , and  $\{W\}$  are obtained by the C-Track w.r.t. its own reference frame,  $\{C\}$ .

In order to calculate the pose error, the following procedures are required. First, the desired pose should be calculated. For demonstration purposes, the robot must insert a cylindrical object (its end-effector) into a precision hole in a workpiece. The clearance between the hole and the cylindrical object is 0.100 mm. When the location of the hole w.r.t. the workpiece is known, the pose of the workpiece frame w.r.t. the workpiece is unknown. We have simply added several retroreflective targets to the workpiece, and the C-Track automatically calculates and assigns a frame associated with these targets. The same is true for the end-effector frame. However, this frame can be modified with the help of HandyProbe (Quebec, Canada), which is a portable, optical-based contact



**Figure 7.** Measured positions (with respect to robot base frame) with corresponding position errors.

inspection tool in Figure 10. HandyProbe allows the user to inspect parts and measure the coordinates of its tip accurately. In addition, the user can create entities and arbitrary reference frames according to the workspace configuration. In this work, the workpiece frame is identified and assigned at the top left corner of the workpiece, as shown in Figure 9. Thus, the position and orientation of the hole can be obtained from the nominal dimensions of the precision artifact. The same procedure is also used for identification of the tool center point (TCP) to define the origin of the tool reference frame.

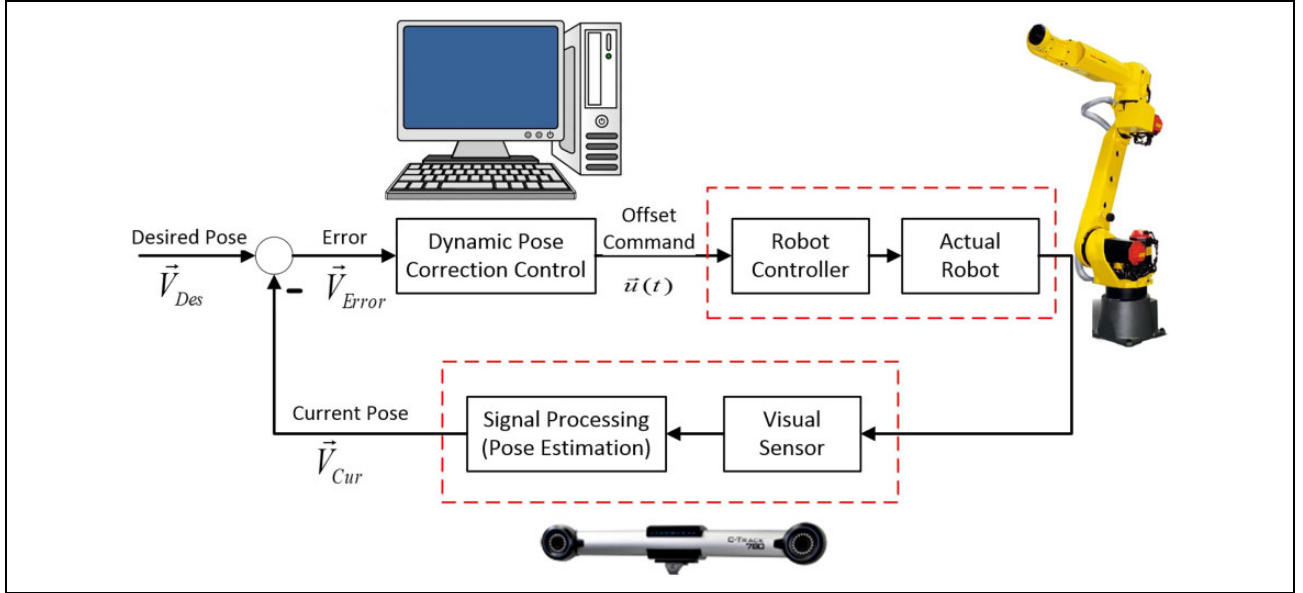
The accumulation of errors in identifying the reference frames with HandyProbe, as well as the errors in the nominal position of the hole w.r.t. the workpiece, explains why the clearance between the hole and the cylindrical object is greater than our accuracy threshold (0.050 mm).

To obtain the desired pose of the tool frame w.r.t. the C-Track frame,  $T_{des}^C \mathbf{H}$ , one has

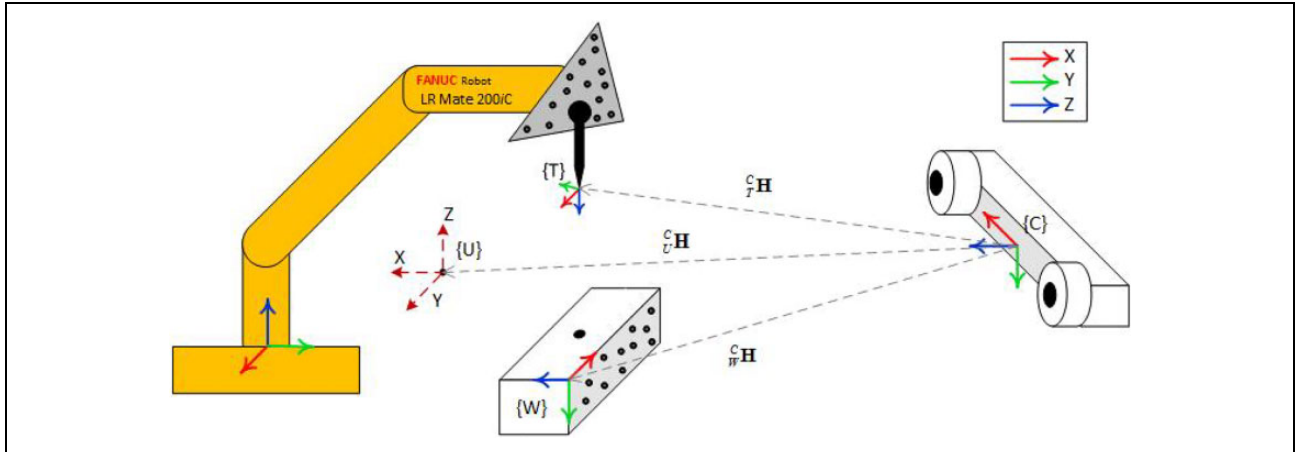
$$T_{des}^C \mathbf{H} = {}_W^C \mathbf{H} \times T_{des}^W \mathbf{H} \quad (6)$$

where  $T_{des}^W \mathbf{H}$  is the homogeneous transformation matrix representing the pose of the robot tool when inserted in the hole with respect to the workpiece frame. The frame is obtained from the computer-aided design model of the workpiece. Matrix  ${}_W^C \mathbf{H}$  represents the pose of the workpiece w.r.t. the C-Track frame and is measured directly. Note that we need to use equation (6) in real time, since the workpiece can move w.r.t. the C-Track (i.e.  ${}_W^C \mathbf{H}$  varies in time) due to vibrations or a moving conveyor of its placement. Now, in both implementations, one needs to specify the pose error w.r.t. the robots user frame, in  $x, y, z$  coordinates for the position, and in  $Z - Y - X$  Euler angles for the orientation. The position error can be obtained by subtracting the actual position vector from the calculated desired one

$${}^C \mathbf{P}_{Error} = T_{des}^C \mathbf{P} - T_{act}^C \mathbf{P} \quad (7)$$



**Figure 8.** Block diagram of the DPC control scheme. DPC: dynamic pose correction.



**Figure 9.** Schematic of the reference frames used for error calculation.

where  ${}^C_{T_{des}} \mathbf{P}$  and  ${}^{C}_{T_{act}} \mathbf{P}$  can be extracted from the homogeneous transformation matrices  ${}^C_{T_{des}} \mathbf{H}$  and  ${}^C_{T_{act}} \mathbf{H}$ , respectively. In order to use these calculations in the robot controller, they should be transformed from the C-Track reference frame to the user frame

$${}^U \mathbf{P}_{Error} = ({}^C_U \mathbf{R})^{-1} \times {}^C \mathbf{P}_{Error} \quad (8)$$

The Euler angles are the orientation errors expressed relative to the user frame and obtaining them is tricky. First, we need to obtain the rotation matrix that corresponds to this orientation error

$${}^{T_{act}}_{T_{des}} \mathbf{R} = ({}^{C}_{T_{act}} \mathbf{R})^{-1} \times {}^{C}_{T_{des}} \mathbf{R} \quad (9)$$

What we are looking for is a sequence of transformations that bring the frame associated with the

actual tool pose,  $\{T_{act}\}$ , to the frame associated with the desired tool pose,  $\{T_{des}\}$ . This sequence should, however, respect the Euler angles convention used by FANUC, that is, one should first rotate  $\{T_{act}\}$  about the  $x$  axis of  $\{U\}$ , then about the  $y$  axis of  $\{U\}$ , and finally about the  $z$  axis of  $\{U\}$ . In another words, one has

$${}^U_{T_{des}} \mathbf{R} = R_z(\gamma) R_y(\beta) R_x(\alpha) \times {}^U_{T_{act}} \mathbf{R} \quad (10)$$

where  $R_x(\alpha)$ ,  $R_y(\beta)$ , and  $R_z(\alpha)$  represent the rotations about the  $x$ ,  $y$ , and  $z$  axis, respectively

$${}^U_{T_{act}} \mathbf{R} = ({}^C_U \mathbf{R})^{-1} \times {}^C_{T_{act}} \mathbf{R} \quad (11)$$

One can also find the desired tool pose w.r.t. the user frame from the following equation



Figure 10. HandyProbe from Creaform.

$$T_{des}^U \mathbf{R} = T_{act}^U \mathbf{R} \times T_{des}^{act} \mathbf{R} \quad (12)$$

From equations (9) to (11), one can find that

$$\begin{aligned} R_z(\gamma)R_y(\beta)R_x(\alpha) &= T_{act}^U \mathbf{R} \times T_{des}^{act} \mathbf{R} \times (T_{act}^U \mathbf{R})^{-1} \\ &= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \end{aligned} \quad (13)$$

where  $r_{ij}$  ( $i = 1, 2, 3$ , and  $j = 1, 2, 3$ ) are known, and  $\alpha$ ,  $\beta$ ,  $\gamma$  are the Euler angles, (or angles  $w$ ,  $p$ , and  $r$ , in FANUC's jargon) that one needs to find. According to the research of Craig,<sup>26</sup> when  $\cos(\beta) \neq 0$ ,  $\alpha$ ,  $\beta$ , and  $\gamma$  can be calculated as follows

$$\begin{aligned} \beta &= a \tan 2 \left( -r_{31}, \sqrt{r_{11}^2 + r_{21}^2} \right) \\ \gamma &= a \tan 2 \left( \frac{r_{21}}{\cos \beta}, \frac{r_{11}}{\cos \beta} \right) \\ \alpha &= a \tan 2 \left( \frac{r_{32}}{\cos \beta}, \frac{r_{33}}{\cos \beta} \right) \end{aligned} \quad (14)$$

Note that since we deal with very small errors, angle  $\beta$  is close to zero, and we do not have problems with the representation singularity that occurs at  $\beta = \pm 90^\circ$ .

After calculating the position and orientation errors (pose error  $\vec{V}_{Error}$ ) w.r.t. the user frame at each iteration, the correction procedure can be implemented. It should be noted that the developed algorithm consists of two major sections: the robot teach-pendant (TP) program and the software that communicates and performs corrections in

real time. The communication between these sections are performed by means of the digital signals (DO[101], DI[101], etc.). Also, the calculations from the C-Track are based on the RMS value of 10 measurements.

**Control scheme.** A PID controller is used to reduce the pose error. The control law is given as follows

$$\vec{u}(t) = K_p \vec{V}_{Error} + K_i \int_0^t \vec{V}_{Error} d\tau + K_d \frac{d}{dt} \vec{V}_{Error} \quad (15)$$

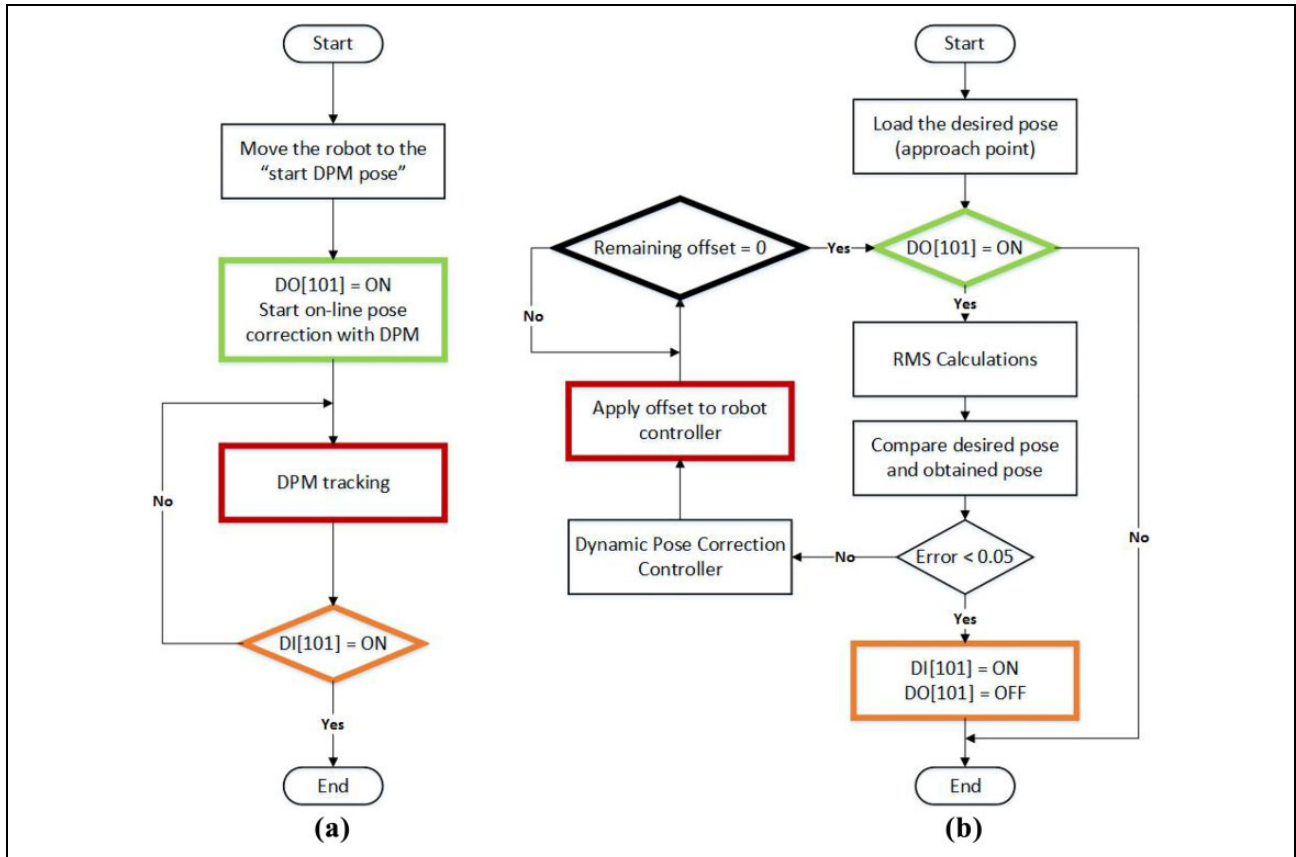
where  $K_p$ ,  $K_i$ , and  $K_d$  are the proportional, integral, and derivative gains, respectively, and have to be tuned to obtain the desired performance of the system,  $\vec{u}(t)$  is the output of DPC scheme which serves as the calculated offsets along  $X$ ,  $Y$ ,  $Z$  and  $W$ ,  $P$ ,  $R$  for the position and orientation, respectively. These offsets are the control commands that are fed to the robot controller.

**DPC with DPM.** In the first approach, the calculated offsets are applied directly to the robot through FANUC's DPM module. This module allows the user to correct the position and orientation of the robot end-effector dynamically w.r.t. the user frame of the robot. Therefore, the calculated offsets are applied to the robot at each iteration, until the required precision is satisfied. The communication between the computer and robot controller uses transmission control protocol/internet protocol.

In order to implement and use the DPM option, it is required to choose the proper DPM mode (according to the application) and define the DPM variables in the software. Also, it is necessary to set offset limits and the rate of the correction on each axis properly for safety issues. In addition, the definition and selection of the synchronization signals to start and finish the online correction should be performed prior to running the TP program. It should be noted that DPM works in Cartesian space only, and it is important to enable and disable DPM at the appropriate points. Some of the key challenges in the implementation of the DPM are fine-tuning of the coefficients for the above-mentioned variables and calculating and applying the corrections quickly. Also, it is important to make sure the corrections are executed by the robot before sending a new correction command. All of these prerequisites and requirements are considered and evaluated in the developed software through a function which initializes the DPM variables according to the user input and sets the offset commands.<sup>25</sup> The flowchart of this implementation is shown in Figure 11.

**DPC without DPM.** Since DPM is an additional software option that must be purchased separately by the user, a second implementation based on updating position register is developed. In this implementation, instead of applying offsets directly to the robot by DPM, the calculated offsets are used to update the destination of the robot in the





**Figure 11.** Flowchart of DPC with DPM: (a) TP program; (b) developed software. DPC: dynamic pose correction; DPM: dynamic path modification; TP: teach-pendant.

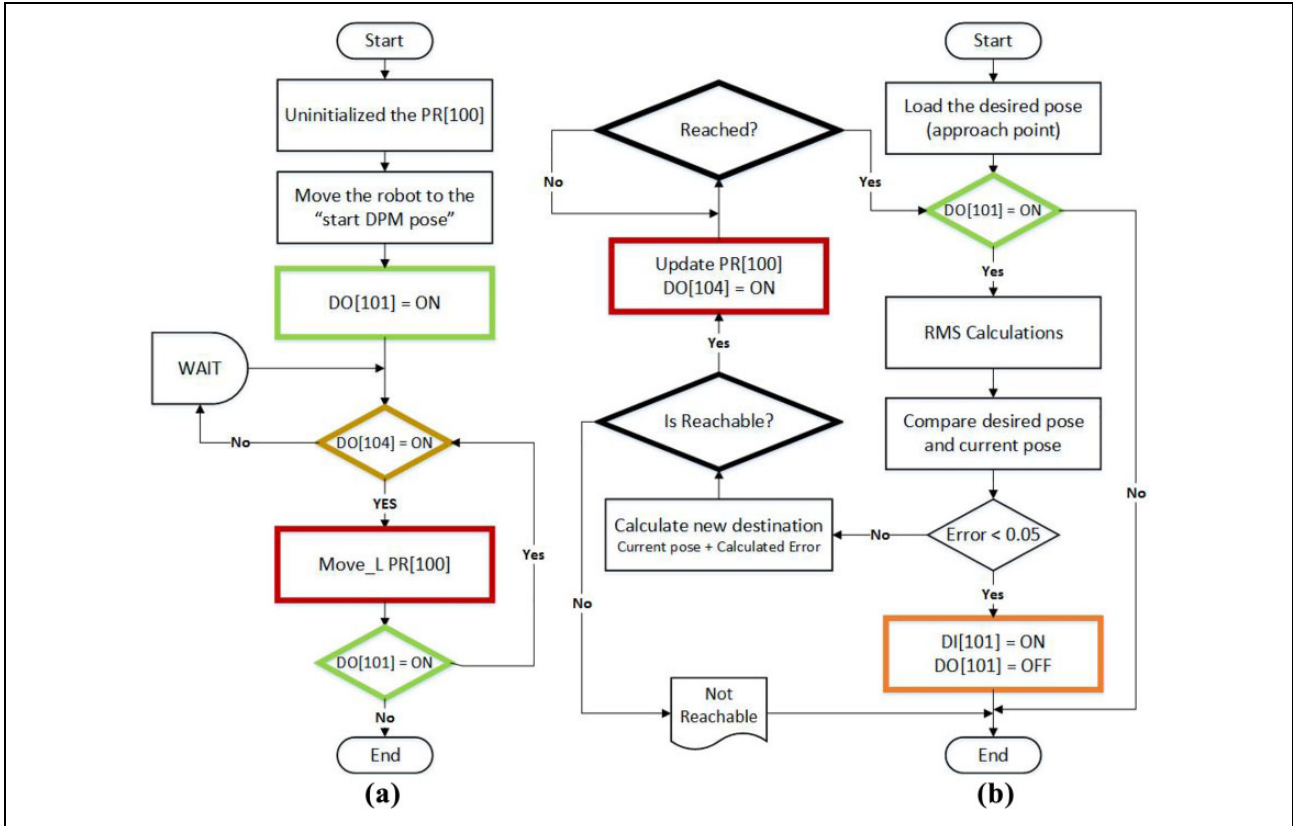
controller. This is done through the developed software and FANUC's personal computer developer's kit (another option, also needed for DPM) by accessing, reading, updating the pose variables, and uploading the new calculated pose information to the robot controller. Figure 12 shows the flowchart of this implementation. Since DPM option is not used in this strategy, it is easier to program than the DPC with DPM module. However, it is still required to define and access the pose variables in the developed software and to consider the pose update rate according to the communication speed and robot motion speed so that it does not interfere with other sections of the program (C-Track measurements and linear movement). Using this implementation, the final pose of the robot after reaching its destination is recorded and stored. Then, the recorded coordinates are compared with the desired ones (already defined in the software). If the error is greater than the accuracy threshold, the recorded coordinates of the robot end-effector is updated and the robot is commanded to move to the new pose in real time. The same procedure repeats until the required precision for the position and orientation are satisfied. All the requirements, including variable definitions, functions, sensor data analysis, and synchronization between software and TP, are taken into account and realized in the developed software.

## Experimental results and verification

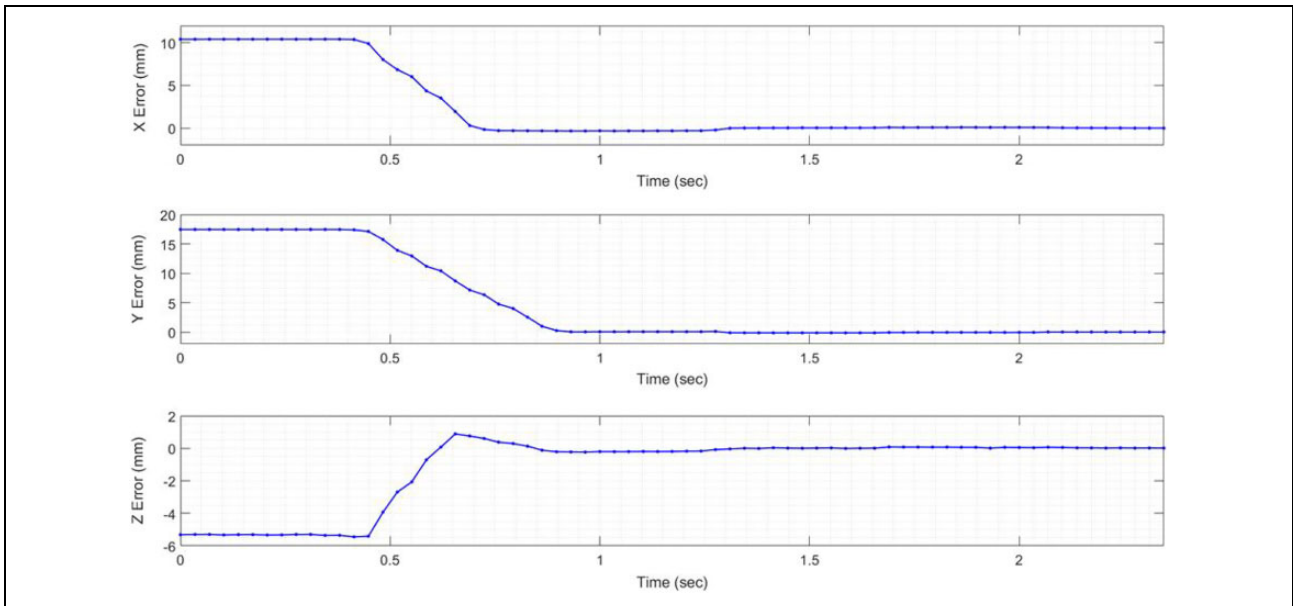
### Tests on FANUC LR Mate 200iC

In order to evaluate the performance of the developed algorithm, the same experimental setup and conditions, including accuracy threshold, DPM mode, correction rate, initial pose, and desired pose are considered. Then, the robot end-effector is moved from an initial pose to a desired pose (not that far from the initial one). The correction process proceeds according to the C-Track information (feedback) until the required precision for the position and orientation is achieved. Figures 13 and 14 show the position and orientation errors for the correction with DPM on FANUC LR Mate 200iC (the implementation presented in Figure 11). The position and orientation errors for updating the pose in the controller on FANUC LR Mate 200iC (the implementation presented in Figure 12) are shown in Figures 15 and 16, respectively.

The experimental results show that by using DPM, the robot can satisfy the required precision after 2.3 s. The second solution, which updates the destination pose of the robot, requires 3.9 s. In order to verify the final accuracy of the proposed algorithm for a single pose correction task, the TriCal is used (Figure 17). Once the robot is at the desired



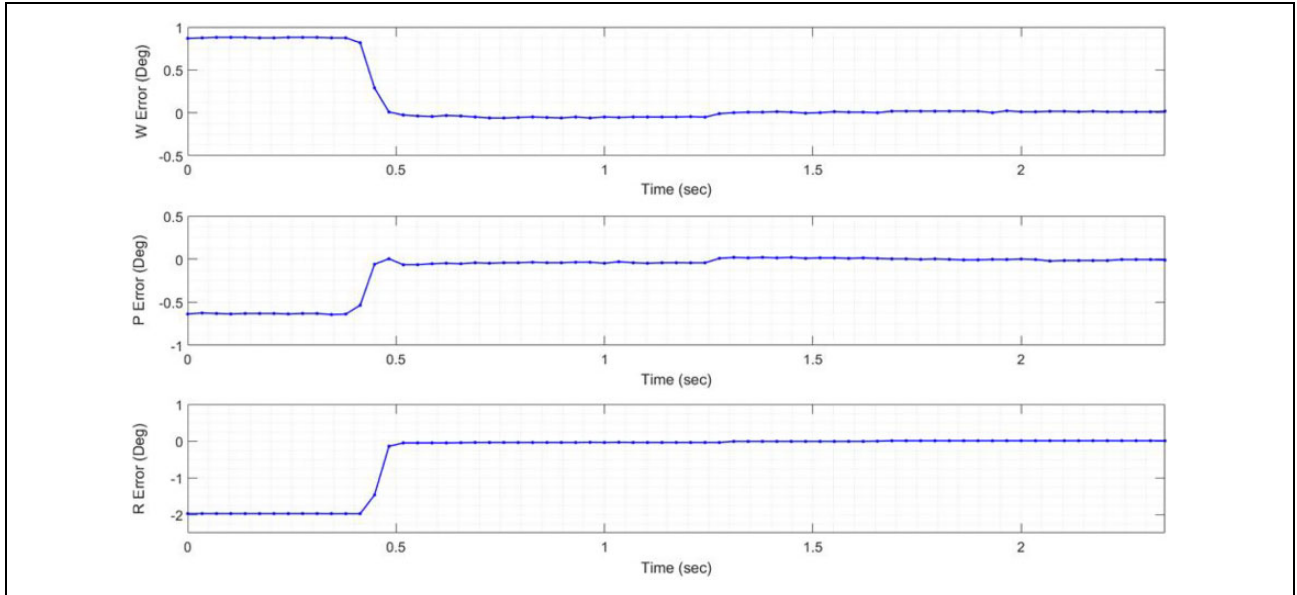
**Figure 12.** Flowchart of DPC without DPM: (a) TP program; (b) developed software. DPC: dynamic pose correction; DPM: dynamic path modification; TP: teach-pendant.



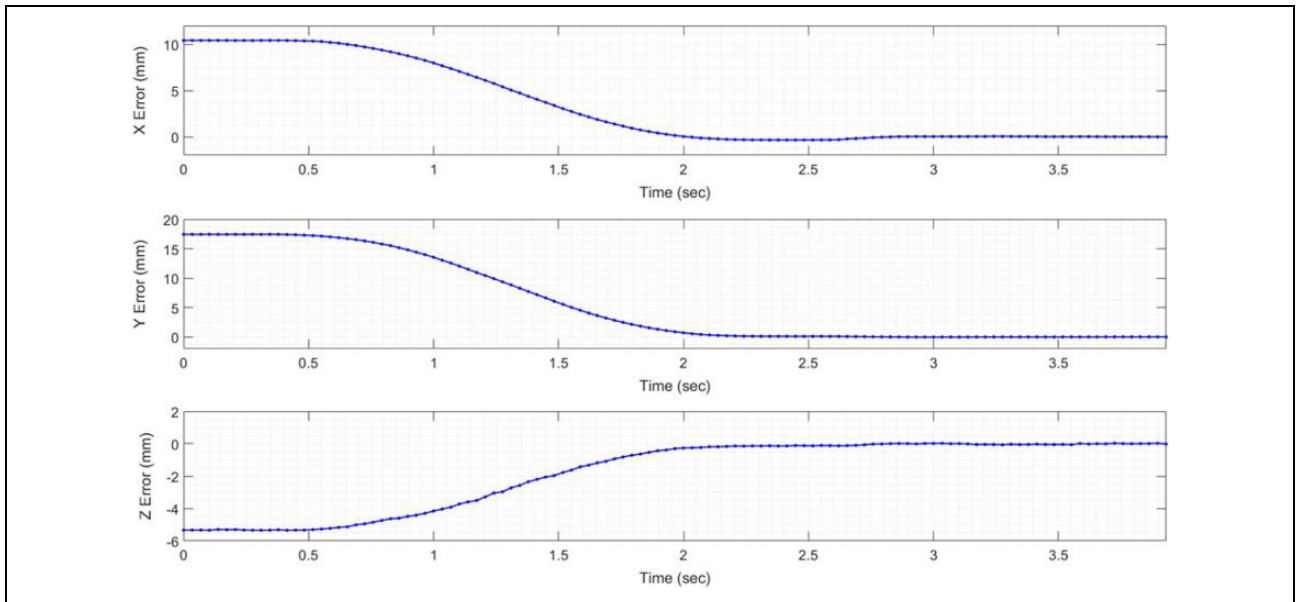
**Figure 13.** Position errors of the end-effector with DPM on FANUC LR Mate 200iC. DPM: dynamic path modification.

pose, the coordinates of the end-effector are recorded by the C-Track and through the developed software. Then, the indicators are set to zero. Next, the robot is moved away and guided to the desired pose by both implementations.

The indicator values are recorded once the robot correction process is terminated. The verification results proved that the final accuracy for single pose correction by both implementations is less than 0.050 mm (<https://youtube/>



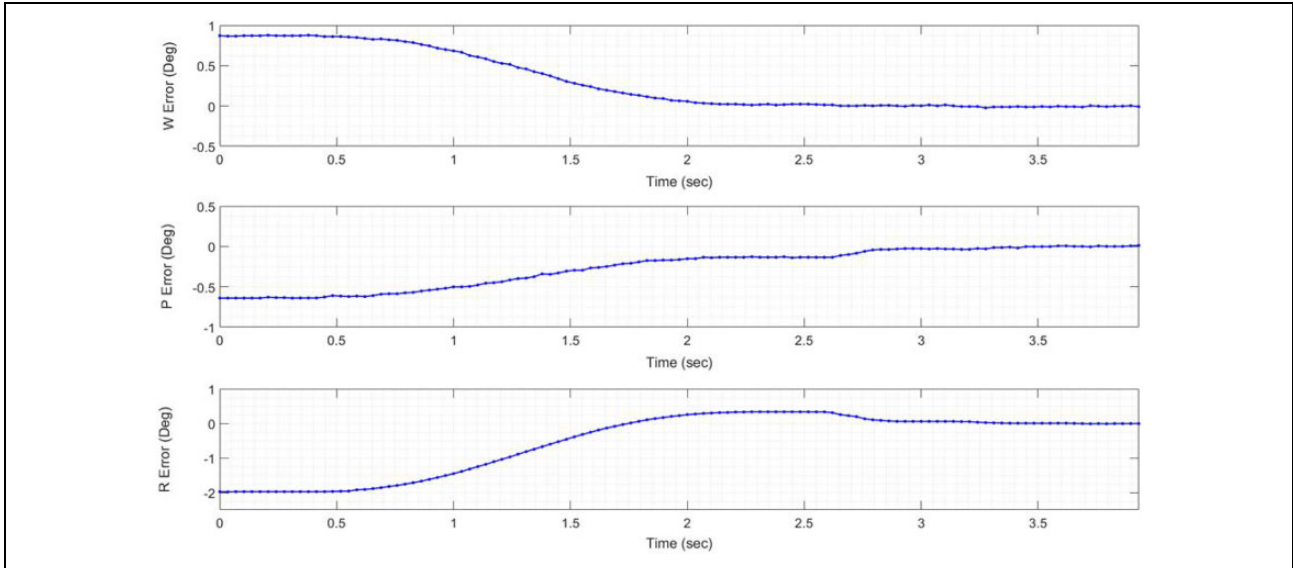
**Figure 14.** Orientation errors of the end-effector with DPM on FANUC LR Mate 200iC. DPM: dynamic path modification.



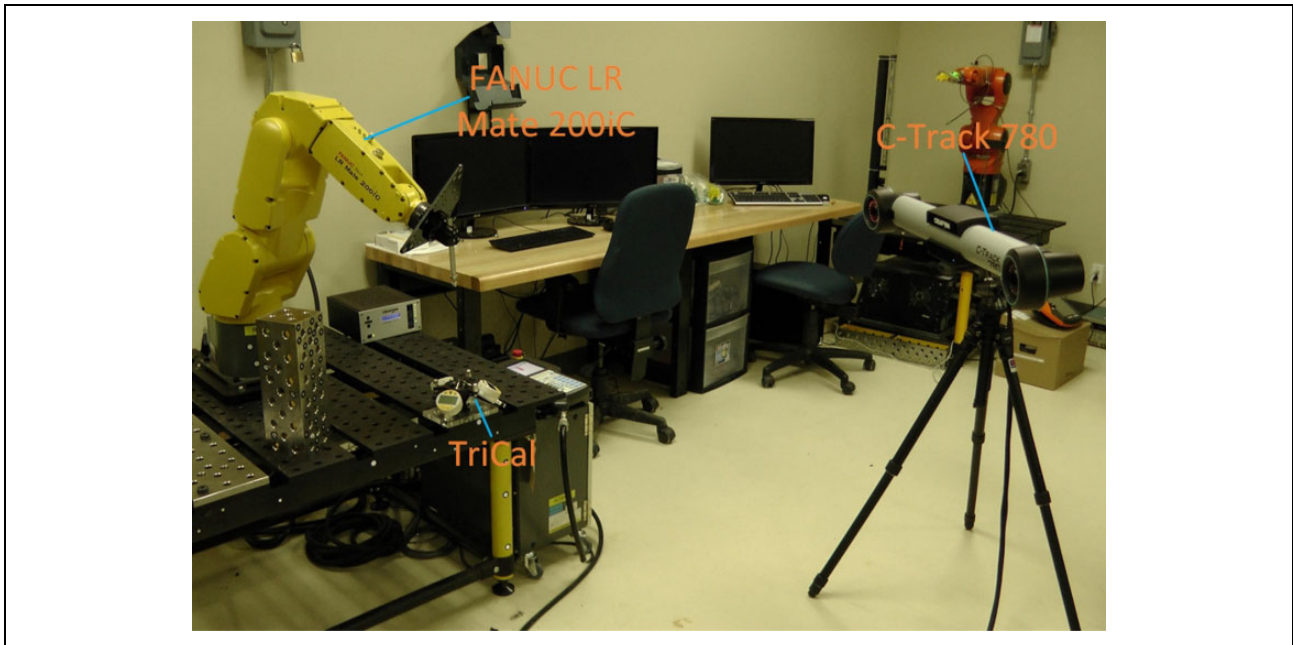
**Figure 15.** Position errors of the end-effector without DPM on FANUC LR Mate 200iC. DPM: dynamic path modification.

JQ60d5z4jwA). Table 1 shows a comparison of two implementations. By comparing them with the conventional static calibration, it is evident that the implementations can reach a significantly higher accuracy. The conventional approach (robot calibration) allows a position accuracy of 0.092 mm for mean errors and 0.240 mm for maximum errors, while the accuracy obtained using the proposed methods is 0.050 mm for maximum error. Also, the proposed algorithm in this article can be implemented on either calibrated or uncalibrated robots since they are independent of the kinematic parameters of the robot. It should

also be noted that both developed implementations can realize a single pose correction. However, in real industrial applications, a program may consist of hundreds of pose information. In that case, in order to perform online pose correction, one needs to follow the above-mentioned procedures for each single pose (which requires correction) and move to the next point when the required precision is satisfied. In other words, the flowcharts shown in Figures 11 and 12 must be implemented for every pose in the TP program. Also, in the first implementation (using DPM option), one needs to activate the DPM at the beginning



**Figure 16.** Orientation errors of the end-effector without DPM on FANUC LR Mate 200iC. DPM: dynamic path modification.



**Figure 17.** Experimental setup for tests on FANUC LR Mate 200iC.

**Table 1.** Comparison between two implementations on FANUC LR Mate 200iC.

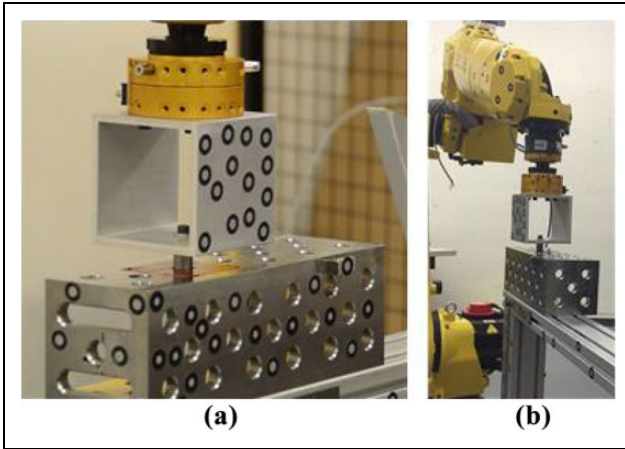
	DPC with DPM					
	X	Y	Z	W	P	R
Initial error	10.5	17.5	-5.5	-0.9	-0.6	-2.0
Response time	2.3 s					
	DPC without DPM					
	X	Y	Z	W	P	R
Initial error	10.5	17.5	-5.5	-0.9	-0.6	-2.0
Response time	3.9 s					

DPC: dynamic pose correction; DPM: dynamic path modification.

**Table 2.** Required licenses for each implementation.

Algorithm \ Licenses	DPC with DPM	DPC by updating PR
VXelements (C-Track)	✓	✓
PCDK (FANUC robot)	✓	✓
DPM (FANUC robot)	✓	✗

DPC: dynamic pose correction; DPM: dynamic path modification; PCDK: personal computer developer's kit; PR: position register.

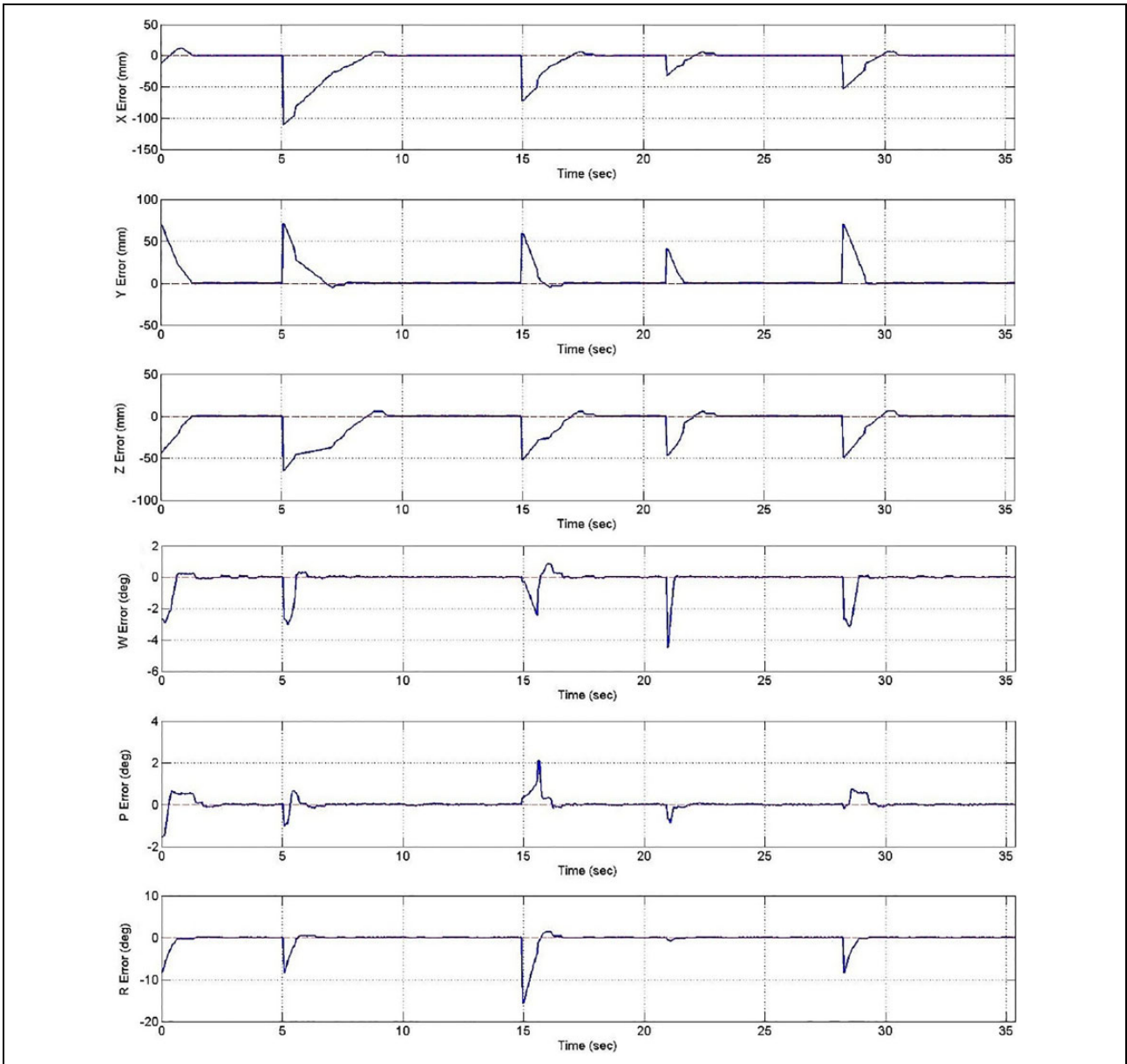


**Figure 18.** Experimental setup for tests on FANUC M20iA.

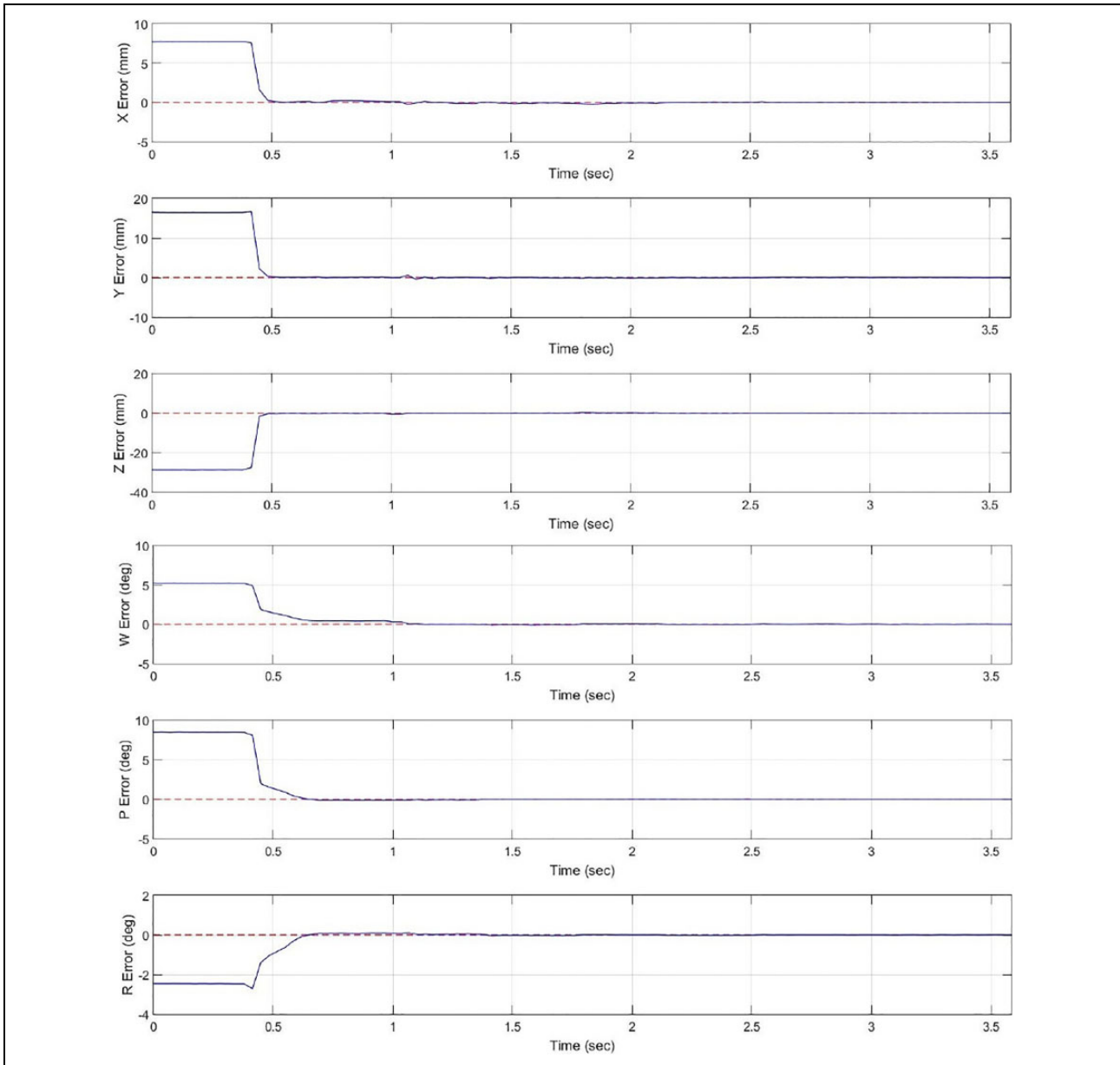
of the program and deactivate it when the program is finished. Tables 1 and 2 show a summary of the comparison between these two implementations and the required software options. Using the first implementation with DPM option, the correction process takes 2.3 s to reach the desired accuracy, and by the second implementation without DPM option, the desired precision is satisfied after 3.9 s. In the experimental tests, the pose accuracy is verified using a C-Track and is compared to that of a conventional calibration approach (i.e. 0.092 mm for mean position error and 0.240 mm for maximum position error).

**Tests on FANUC M20iA**

The DPC with DPM option is also implemented on the other industrial robot, FANUC M20iA. The experimental



**Figure 19.** Pose errors of the end-effector with DPM on FANUC M20iA when it inserts the tool into a hole on a cube. DPM: dynamic path modification.



**Figure 20.** Pose errors of the end-effector with DPM on FANUC M20iA. DPM: dynamic path modification

setup of the implementation is shown in Figure 18. First, the end-effector is jogged to the desired pose (above the hole on the cube), and the relation between the tool model and cube model is recorded in the developed software as a homogeneous transformation matrix. Then, the end-effector moves from an initial pose toward the desired pose according to the sensor measurements by applying online corrections. The real-time corrections provided by the developed algorithm continue until the error between the pose of the tool model (provided by the C-Track) and desired pose (already stored in the DPC software) satisfies the required precision. When the precision is satisfied, the robot will insert the tool into the hole and extract (Figure 18). Furthermore, the developed DPC algorithm is capable

of recording and automatically updating the desired pose of the robot end-effector according to the configuration of the cube model.

In order to validate the performance of the developed algorithm, the correction process is performed continuously with different configurations of the cube in the C-Track's field of view. The experimental results are presented in Figure 19. As shown in this figure, the initial error on each axis is being compensated by the developed DPC algorithm, and the robot is guided to the desired pose. The pose correction process continues until the required precisions for the position and orientation (0.050 mm and 0.050°) are satisfied. Then, the configuration of the cube is changed, and the desired pose of the robot is updated in the

**Table 3.** DPC performance on FANUC M20iA.

	Position			Orientation		
	X	Y	Z	W	P	R
Initial error	7.5	17.5	-28.0	6.0	8.0	-2.5
Response time	3.6 s					

DPC: dynamic pose correction.

software. Therefore, the initial errors are changed after each insertion (for instance at  $t = 5$  s). According to the updated desired pose, the robot is guided to the same hole by the developed DPC algorithm (<https://youtu.be/ABzA7fRc8T8>).

The experimental results demonstrate that the robot can follow the hole and satisfy the required precision based on the real-time pose information of the cube model. Figure 20 shows the experimental results for a single pose correction. In this experiment, the correction is performed only once to evaluate the response of the system using the developed DPC algorithm. The initial error, response time, and the number of corrections to achieve the desired accuracy are shown in Table 3. According to this table, the distance error (approximately 34.0 mm) as well as the orientation error are corrected and the precision threshold is met after approximately 3.6 s.

## Conclusion

In this article, a dual-camera pose measurement sensor (C-Track 780) was used for online correction purposes. A DPC algorithm was proposed to improve the accuracy of a 6-DOF serial industrial robots for stationary tasks. The developed DPC algorithm had been implemented with two options. The implementation using the DPM option was tested on FANUC LR Mate 200iC and FANUC M20iA, which applies dynamic corrections directly to the robot. The implementation without the DPM option was tested on FANUC LR Mate 200iC. It was carried out by updating the absolute pose of the robot end-effector in the controller according to the C-Track measurements. All experimental tests demonstrate that both implementations could improve the positioning accuracy of the robots to 0.050 mm for position and 0.050° for orientation. However, the DPC without DPM took a longer time to complete the same task. Moreover, the DPC with DPM on FANUC M20iA was slower than that on FANUC LR Mate 200iC since FANUC M20iA had a higher payload. Compared with the results of static calibration on FANUC LR Mate 200iC, where the robot mean error is 0.092 mm and the maximum position error is 0.240 mm, the DPC has improved the pose accuracy greatly.

In the future, we will apply the proposed DPC to other types of industrial robots such as ABB, KUKA, and so on, and will consider the contact force stabilization in addition to pose correction for the potential applications such as tasks of riveting, drilling, and welding.

## Acknowledgements

The authors would like to thank NSERC, CRIAQ, Creaform, GE Aviation, and Coriolis Composites for their support to this work.


## Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

## Funding

The author(s) disclosed the receipt of following financial support for the research, authorship, and/or publication of this article: This work was supported by Natural Sciences and Engineering Research Council of Canada (NSERC, Manu 602), CRIAQ, Creaform, GE Aviation, and Coriolis Composites.

## ORCID iD

Tingting Shu  <http://orcid.org/0000-0002-1213-0609>

## References

- Jiang Y, Jiang Y, Huang X, et al. An on-line compensation method of a metrology-integrated robot system for high-precision assembly. *Ind Rob Int J* 2016; 43(6): 647–656.
- Alici G and Shirinzadeh B. A systematic technique to estimate positioning errors for robot accuracy improvement using laser interferometry based sensing. *Mech Mach Theory* 2005; 40: 879–906.
- Greenway B. Robot accuracy. *Ind Rob* 2000; 27(4): 257–265.
- Mooring BW, Roth ZS, and Driels MR. *Fundamentals of manipulator calibration*. Wiley-interscience, 1991.
- Elatta A, Gen LP, Zhi FL, et al. An overview of robot calibration. *Inf Technol J* 2004; 3(1): 74–78.
- Joubair A. *Contribution à l'amélioration de la précision absolue des robots parallèles*. PhD Thesis, École de technologie supérieure, 2012.
- Motta JMS, de Carvalho GC, and McMaster R. Robot calibration using a 3D vision-based measurement system with a single camera. *Rob Comput Integr Manuf* 2001; 17: 487–497.
- Song Y, An G, and Zhang J. Positioning accuracy of a medical robotic system for spine surgery. In: *2nd international conference on biomedical engineering and informatics (BMEI'09)* (eds P Qiu, C Yiu, H Zhang, and X Wen), Tianjin, China, 17–19 October 2009, pp. 1–5, IEEE.
- Joubair A, Nubiola A, and Bonev I. Calibration efficiency analysis based on five observability indices and two calibration models for a six-axis industrial robot. *SAE Int J Aerosp* 2013; 6(1): 161–168.
- Nubiola A, Slamani M, Joubair A, et al. Comparison of two calibration methods for a small industrial robot based on an optical cmm and a laser tracker. *Robotica* 2014; 32(3): 447–466.
- Gaudreault M, Joubair A, and Bonev IA. Local and closed-loop calibration of an industrial serial robot using a new low-cost 3D measuring device. In: *Robotics and automation (ICRA), IEEE international conference* (ed A Okamura),

- Stanford University, Stockholm, Sweden, 16–21 May 2016, pp. 4312–4319. IEEE.
12. Joubair A and Bonev IA. Kinematic calibration of a six-axis serial robot using distance and sphere constraints. *Int J Adv Manuf Technol* 2015; 77(1–4): 515–523.
  13. Hutchinson S, Hager GD, and Corke PI. A tutorial on visual servo control. *IEEE Trans Robot Autom* 1996; 12(5): 651–670.
  14. Chaumette F and Hutchinson S. Visual servo control. I. Basic approaches. *IEEE Robot Autom Mag* 2006; 13(4): 82–90.
  15. Bone GM and Capson D. Vision-guided fixtureless assembly of automotive components. *Robot Comput Integr Manuf* 2003; 19(1–2): 79–87.
  16. Lertpiriyasuwat V and Berg MC. Adaptive real-time estimation of end-effector position and orientation using precise measurements of end-effector position. *IEEE/ASME Trans Mechatron* 2006; 11: 304–319.
  17. Norman AR, Schönberg A, Grolach IA, et al. Validation of iGPS as an external measurement system for cooperative robot positioning. *Int J Adv Manuf Technol* 2013; 64(1–4): 427–446.
  18. Jin Z, Yu C, Li J, et al. A robot assisted assembly system for small components in aircraft assembly. *Ind Rob Int J* 2014; 41: 413–420.
  19. Posada JD, Schneider U, Pidan S, et al. High accurate robotic drilling with external sensor and compliance model-based compensation. In: *Robotics and automation (ICRA), IEEE international conference* (ed A Okamura), Stanford University, Stockholm, Sweden, 16–21 May 2016, pp. 3901–3907. IEEE.
  20. José de Jesús R. Discrete time control based in neural networks for pendulums. *Appl Soft Comput* 2018; 68: 821–832.
  21. Pan Y, Guo Z, Li X, et al. Output-feedback adaptive neural control of a compliant differential SMA actuator. *IEEE Trans Control Syst Technol* 2017; 25: 2202–2210.
  22. Nikon. High accuracy robotics. [http://www.nikonmetrology.com/en\\_US/Products/In-process-measurement/Adaptive-Robot-Control](http://www.nikonmetrology.com/en_US/Products/In-process-measurement/Adaptive-Robot-Control) (2016, accessed 15 March 2016).
  23. Hsiao T and Huang PH. Iterative learning control for trajectory tracking of robot manipulators. *Int J Autom Smart Technol* 2017; 7: 133–139.
  24. Brethé JF and Dakyo B. A stochastic ellipsoid approach to repeatability modelisation of industrial manipulator robots. In: *Intelligent robots and systems, IEEE/RSJ international conference*, EPLF Lausanne, Switzerland, 30 September–3 October 2002, Vol. 2, pp. 1608–1613. IEEE.
  25. Gharaaty S. *Accuracy enhancement of industrial robots by dynamic pose correction*. Master's Thesis, Concordia University, Canada, 2016.
  26. Craig JJ. *Introduction to robotics: mechanics and control*, vol. 3. NJ, USA: Pearson/Prentice Hall Upper Saddle River, 2005.