

Article

An Improved Data-Driven Calibration Method with High Efficiency for a 6-DOF Hybrid Robot

Zhibiao Yan ¹, Youyu Wang ², Haitao Liu ^{1,*} , Juliang Xiao ¹ and Tian Huang ¹

¹ Key Laboratory of Modern Mechanisms and Equipment Design of the State Ministry of Education, Tianjin University, Tianjin 300072, China

² Beijing Institute of Spacecraft System Engineering CAST, Beijing 100094, China

* Correspondence: liuht@tju.edu.cn

Abstract: This paper proposes an improved data-driven calibration method for a six degrees of freedom (DOF) hybrid robot. It focuses mainly on improving the measurement efficiency and practicability of existing data-driven calibration methods through the following approaches. (1) The arbitrary motion of the hybrid robot is equivalently decomposed into three independent sub-motions by motion decomposition. Sequentially, the sub-motions are combined according to specific motion rules. Then, a large number of robot poses can be acquired in the whole workspace via a limited number of measurements, effectively solving the curse of dimensionality in measurement. (2) A mapping between the nominal joint variables and joint compensation values is established using a back propagation neural network (BPNN), which is trained directly using the measurement data through a unique algorithm involving inverse kinematics. Thus, the practicability of data-driven calibration is significantly improved. The validation experiments are carried out on a TriMule-200 robot. The results show that the robot's maximal position/orientation errors are reduced by 91.16%/88.17% to 0.085 mm/0.022 deg, respectively, after calibration.

Keywords: calibration; hybrid robot; data-driven; neural network



Citation: Yan, Z.; Wang, Y.; Liu, H.; Xiao, J.; Huang, T. An Improved Data-Driven Calibration Method with High Efficiency for a 6-DOF Hybrid Robot. *Machines* **2023**, *11*, 31. <https://doi.org/10.3390/machines11010031>

Academic Editor: Dan Zhang

Received: 21 October 2022

Revised: 21 December 2022

Accepted: 22 December 2022

Published: 27 December 2022



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Due to the high stiffness, large workspace/footprint ratio, and desirable dynamic characteristics, hybrid robots have been widely used in modern high-end equipment manufacturing fields such as aerospace, military, and rail transit [1–3]. Pose accuracy is the most important index to evaluate the machining performance of hybrid robots. On the premise of guaranteeing good repeatability through manufacturing and assembly, calibration is an effective way to improve the robot pose accuracy [4–7].

Model-based calibration, also named kinematic calibration, is a classical technique widely used in engineering applications. It usually consists of four steps: modeling, measurement, identification, and compensation. The core step is to establish an error model that satisfies the requirements of continuity, completeness, and minimization [8,9]. However, a robot has many error sources, including not only geometric errors such as assembly and manufacturing errors [10], but also many non-geometric errors such as friction, backlash, thermal deformation, and flexible deformation [11]. Moreover, the study [12] has shown that almost 20% of robot errors are caused by error sources that vary with the robot configurations, such as straightness errors in moving pairs and pitch errors. Hence, establishing a complete model that considers all error sources seems to be an impossible challenge.

The limitation of model-based calibration incentivizes the application of data-driven methodologies [13], which estimate errors with an arbitrary posture using pre-measured errors at certain locations by curve fitting [14], spatial interpolation [15–17], and artificial neural networks (ANNs) [18–20]. Then, compensation is conducted based on the estimated

values [21]. It usually involves three steps: measurement, estimation, and compensation. Along this track, Alici et al. proposed an approach to estimate the robot position errors using ordinary polynomials [22] and Fourier polynomials [23]. Bai et al. presented fuzzy interpolation to estimate the position errors using the pre-measured errors on cubic lattices [24]. According to the spatial similarity of position errors, Tian et al. [25] introduced an inverse distance weighting (IDW) method. Cai et al. [26] utilized the kriging algorithm to approximate the robot error surface. Analogously, ANNs, such as extreme learning machine (ELM) [27], back-propagation neural network (BPNN) [28,29], and radial basis function neural network (RBFNN) [30] have been gradually applied in data-driven calibration. Although the data-driven method seems to be a promising approach due to its excellent prediction performance, it still cannot ignore the limitations in measurement and compensation, which seriously restricts its further development and application.

Data-driven calibration can theoretically compensate for all errors in the robot system. However, its compensation accuracy relies on massive measurement data. The study [31] has shown that a few sampling configurations cannot accurately describe the robot error distribution, leading to poor accuracy after calibration, and the accuracy of some configurations may be even worse than that before calibration. It is worth mentioning that the measurement task always takes up the most of time in the calibration. Especially for robots with high-dimensional joint space, the sampling data required for accurate calibration exponentially increase with increasing dimensions of the joint space, leading to the curse of dimensionality [32]. Consequently, the measurement task might take several days. The limitation of low measurement efficiency is an urgent problem to be solved in data-driven calibration.

Due to the limitation of the openness of the robot system and the difficulty in solving the analytic inverse kinematics, indirect approaches [33] that modify the joint input or Cartesian coordinates are widely used in error compensation. Chen et al. [34,35] directly took the estimated error E (the deviation between the measured pose P_m and the desired pose P_n ; i.e., $E = P_m - P_n$) as the compensation value and assumed that commanding a robot to move to the Cartesian position $P_n - E$ would cause it to reach P_n in reality; i.e., $P_m = P_n$. However, the robot will actually reach a new pose, $P_m = P_n - E + E^*$, which involves an error E^* of another pose $P_n - E$, where in the general case $E \neq E^*$. To acquire the optimal compensation value, Zhao et al. proposed a series of iterative algorithms based on the kinematics [36], Jacobian [37], and optimization algorithms such as particle swarm optimization (PSO) [38] and genetic algorithm (GA) [39]. Although the compensation accuracy can be guaranteed to a certain extent, these approaches are not feasible for online compensation because of the high computational burden. In addition, the compensation approaches mentioned above are carried out based on the estimated errors, which inevitably introduce estimation residuals. More importantly, the cumbersome calibration process of prediction and compensation is very time-consuming, preventing this approach from being used in field applications.

Aiming to resolve these problems, a new calibration method has been proposed in research [40], which could effectively solve the contradiction between calibration accuracy and measurement efficiency, as well as the problem of real-time error compensation. However, it still has some shortcomings in practical application, as described in detail in Sections 3.1 and 4.1. Thus, an improved data-driven methodology is investigated in this paper. The significance lies in the improvement of measurement efficiency and practicality. In the measurement, an arbitrary motion of the robot is equivalently decomposed into three independent sub-motions. Next, the sub-motions are combined according to specific motion rules. In this manner, a large number of robot poses can be acquired in the whole workspace via a limited number of measurements, which effectively solves the curse of dimensionality in measurement. In compensation, the mapping between the nominal joint variables and joint compensation values is established based on a BPNN, which is trained directly using the measurement data through a unique training algorithm involving the robot inverse kinematics. Thus, the practicability of the data-driven methodology is greatly improved.

The paper is organized as follows. Section 2 is the system description of the six degrees of freedom (DOF) hybrid robot. Section 3 introduces the principle of motion decomposition and the implementation process of decomposition measurement. Section 4 illustrates an improved data-driven calibration methodology from the aspects of the calibration principle, mapping model designing, and network training. The validation experiments are carried out on a TriMule-200 robot in Section 5. Finally, conclusions are drawn in Section 6.

2. System Description

Figure 1 shows the 3D model of the TriMule robot. It mainly consists of a 3-DOF parallel mechanism and a 3-DOF wrist. The 1T2R parallel mechanism is composed of a 6-DOF UPS limb and a 2-DOF planar parallel mechanism involving two actuated RPS limbs and a passive RP limb. The planar parallel mechanism is connected to the base frame through a pair of R joints. The wrist is an RRR series mechanism with three axes intersecting at a common point. Here, R, U, S, and P represent the revolute joint, universal joint, spherical joint, and prismatic joint, respectively. The underlined R and P represent the actuated revolute joint and prismatic joint, respectively.

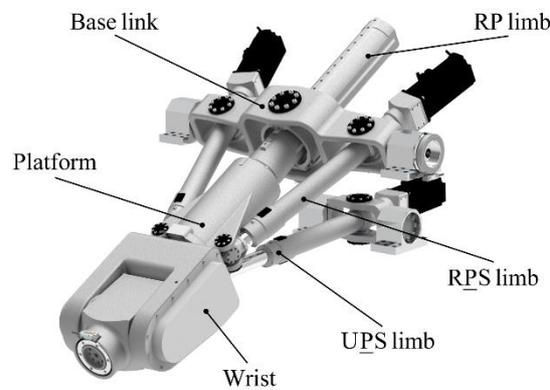


Figure 1. Three-dimensional model of the TriMule robot.

As illustrated in Figure 2, we number the UPS limb and two actuated RPS limbs as limbs 1, 2, and 3, respectively. The passive RP limb together with the wrist is limb 4. B_1 is the center of the U joint of limb 1; $B_i (i = 2, 3)$ is the intersections of the R joints of limb i and the rotation axis of the planar parallel mechanism defined by the R joints connecting to the base frame; and $A_i (i = 1, 2, 3)$ is the center of the S joint of limb i . Let P be the intersection of three orthogonal axes of the wrist and C be the tool center point (TCP) of the robot end-effector. The base frame \mathcal{K} is placed at point O , the intersection of the R joint of the RP limb and the rotation axis of the planar parallel mechanism, with its z -axis normal to $\Delta B_1 B_2 B_3$, the triangle with vertices at points B_i , and its x -axis coincident with $\vec{B_3 B_2}$.

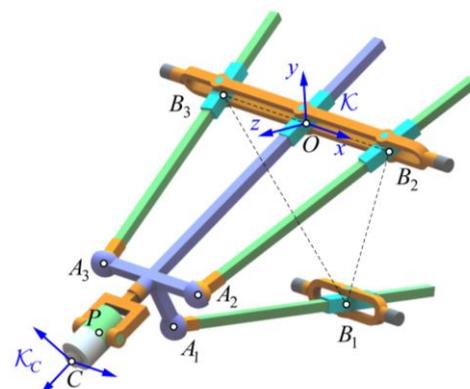


Figure 2. Schematic diagram of the TriMule robot.

3. Motion Decomposition and Measurement

This section focuses on the decomposition measurement for a 6-DOF hybrid robot. It involves the principle of motion decomposition and the implementation process of decomposition measurement.

3.1. Principle of Motion Decomposition

Sufficient pose measurement data are the basis of data-driven calibration and are crucial to a good compensation effect. However, the measurement data required by accurate calibration exponentially increase with increasing dimensions of the joint space. The curse of dimensionality is the main challenge in measurement tasks. Decomposition measurement is an effective approach to solve this problem. In previous research [40], we decomposed a 5-DOF hybrid robot into a 2-DOF wrist and a 3-DOF parallel mechanism through mechanism decomposition. The pose errors of the two substructures are measured respectively and then composed to obtain those of the hybrid robot, effectively improving the measurement efficiency. However, the pose errors of the parallel mechanism can hardly be measured directly due to the occlusion of the wrist. An additional transfer frame must be established to connect the two substructures so that indirect measurement can be completed with the help of the end effector. As a result, it involves not only transformation errors but also a heavy burden in the measurement.

To solve these troubles, we proposed a motion decomposition method in this paper. The arbitrary motion of the robot is equivalently decomposed into three independent sub-motions. The sub-motions are achieved by driving the actuated joints according to specific rules. Consequently, decomposition measurement can be realized by only measuring the end pose of the robot, which is more practical and efficient.

According to the mechanism characteristics of TriMule, we partition the joint variables $\mathbf{q} = (q_1, q_2, \dots, q_6)$ into two subsets, $\mathbf{u} = (q_1, q_2, q_3)$ and $\mathbf{v} = (q_4, q_5, q_6)$, where \mathbf{u} contains the joint variables of the parallel mechanism and \mathbf{v} contains the joint variables of the wrist. The mappings between the robot joint space and the operation space are defined as the kinematics functions $K_i: \mathbb{R}^6 \rightarrow SE(3)$, and the corresponding motions can be expressed by the homogeneous transformation matrices T_i . Hence, the forward kinematic function of the hybrid robot $K(\mathbf{q})$ can be written as:

$$K(\mathbf{q}) = K(\mathbf{u}, \mathbf{v}) = K_u(\mathbf{u}) \cdot K_v(\mathbf{v}) \quad (1)$$

where K_v and K_u are the kinematic functions of the wrist and the parallel mechanism defined by \mathbf{v} and \mathbf{u} , respectively.

Then, Equation (1) can be equivalently transformed into:

$$K(\mathbf{q}) = K_u(\mathbf{u}) \cdot K_v(\tilde{\mathbf{v}}) \cdot K_v(\tilde{\mathbf{v}})^{-1} \cdot K_u(\tilde{\mathbf{u}})^{-1} \cdot K_u(\tilde{\mathbf{u}}) \cdot K_v(\mathbf{v}) \quad (2)$$

where $\tilde{\mathbf{u}}$ is an arbitrarily fixed value for \mathbf{u} , $K_u(\tilde{\mathbf{u}})$ represents the motion of the parallel mechanism associated with $\tilde{\mathbf{u}}$, $\tilde{\mathbf{v}}$ is an arbitrarily fixed value for \mathbf{v} , and $K_v(\tilde{\mathbf{v}})$ represents the motion of the wrist associated with $\tilde{\mathbf{v}}$.

Finally, substituting Equation (1) into Equation (2) results in the motion decomposition formula:

$$K(\mathbf{q}) = K(\mathbf{u}, \tilde{\mathbf{v}}) \cdot K(\tilde{\mathbf{u}}, \tilde{\mathbf{v}})^{-1} \cdot K(\tilde{\mathbf{u}}, \mathbf{v}) = K(\mathbf{q}_1) \cdot K(\mathbf{q}_0)^{-1} \cdot K(\mathbf{q}_2) \quad (3)$$

where $\mathbf{q}_1 = (\mathbf{u}, \tilde{\mathbf{v}})$, $\mathbf{q}_0 = (\tilde{\mathbf{u}}, \tilde{\mathbf{v}})$, and $\mathbf{q}_2 = (\tilde{\mathbf{u}}, \mathbf{v})$.

Hence, an arbitrary motion $K(\mathbf{q})$ of the hybrid robot can be equivalently decomposed into three independent motions: $K(\mathbf{q}_0)$, $K(\mathbf{q}_1)$, and $K(\mathbf{q}_2)$.

3.2. Decomposition Measurement and Composition

According to the definition of the kinematics functions K_i , Equation (3) can be rewritten as

$$T = T_1 T_0^{-1} T_2 \quad (4)$$

where T , T_0 , T_1 , and T_2 represent the poses of the hybrid robot at different configurations q , q_0 , q_1 , and q_2 , respectively.

Thus, for a certain motion $K(q)$, except for the direct measurement at configuration q , we can also measure the poses of the hybrid robot T_0 , T_1 , and T_2 at another three specific configurations, q_0 , q_1 , and q_2 , respectively, and then obtain the required pose T through composition. It is worth noting that q_0 is an arbitrarily fixed value for q , and q_1 , q_2 are related to q_0 and q .

Although the decomposition measurement demonstrates no obvious advantages for a single configuration and even increases the measurement task, it has great advantages in terms of measurement efficiency for massive measurement tasks. In addition to Figure 3, the detailed steps of the method are described as follows. For convenience of description, the stationary joints of the hybrid robot are marked in red, and the moving joints are marked in green.

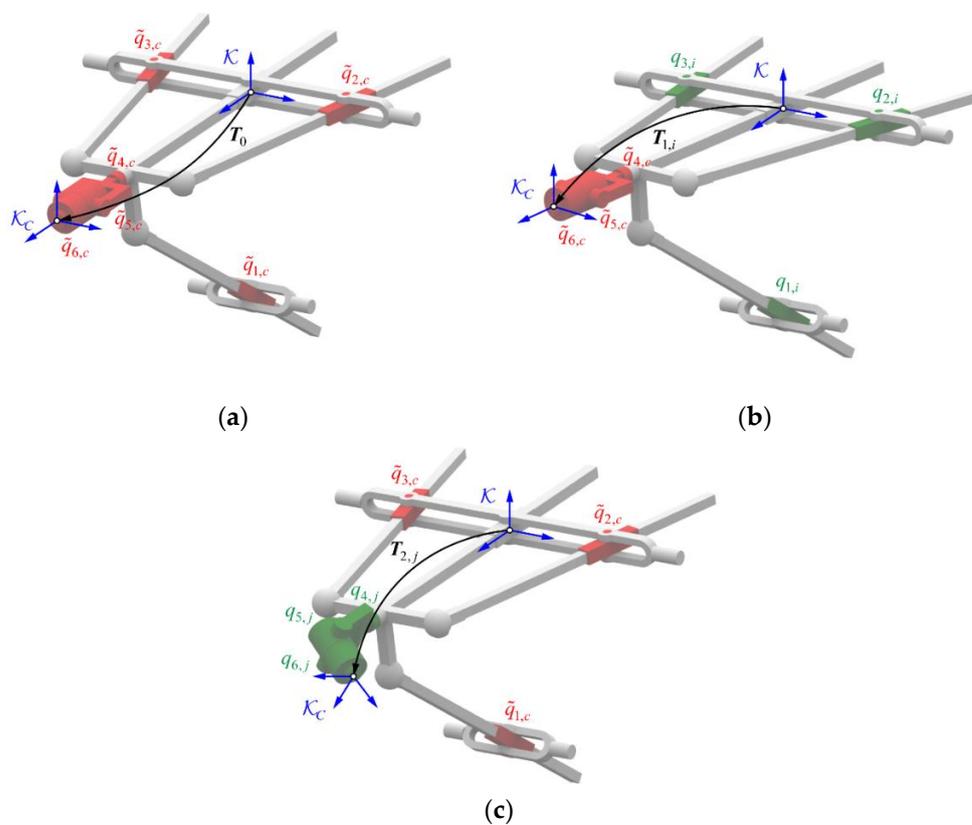


Figure 3. Schematic diagram of decomposition measurement: (a) Reference configuration; (b) Motion of the parallel mechanism; and (c) Motion of the wrist.

(1) The base frame \mathcal{K} is established as defined in Section 2, and an end frame \mathcal{K}_C is set up at the robot end-effector (see Figure 3). All measurements hereinafter are conducted in the base frame \mathcal{K} , and T_* denotes the transformation matrix of frame \mathcal{K}_C with respect to frame \mathcal{K} for different configurations q_* .

(2) An arbitrary fixed value $\tilde{q}_0 = (\tilde{u}, \tilde{v}) = (\tilde{q}_{1,c}, \tilde{q}_{2,c}, \tilde{q}_{3,c}, \tilde{q}_{4,c}, \tilde{q}_{5,c}, \tilde{q}_{6,c})$ is set as the reference configuration. The hybrid robot is moved to \tilde{q}_0 , and its end pose T_0 is measured, as shown in Figure 3a.

(3) The m measurement configurations $\mu_i = (q_{1,i}, q_{2,i}, q_{3,i}), i = 1, 2, \dots, m$ are uniformly selected in the joint space of the parallel mechanism. The hybrid robot is moved to each configuration $q_{1,i} = (u_i, \tilde{v}) = (q_{1,i}, q_{2,i}, q_{3,i}, \tilde{q}_{4,c}, \tilde{q}_{5,c}, \tilde{q}_{6,c})$ successively by keeping the wrist stationary at the reference configuration and moving the parallel mechanism independently. Next, the end pose of the robot $T_{1,i} (i = 1, 2, \dots, m)$ is measured at each configuration, as shown in Figure 3b.

(4) The n measurement configurations $v_j = (q_{4,j}, q_{5,j}, q_{6,j}), j = 1, 2, \dots, n$ are uniformly selected in the joint space of the wrist. The hybrid robot is moved to each configuration $q_{2,j} = (\tilde{u}, v_j) = (\tilde{q}_{1,c}, \tilde{q}_{2,c}, \tilde{q}_{3,c}, q_{4,j}, q_{5,j}, q_{6,j})$ successively by keeping the parallel mechanism stationary at the reference configuration and moving the wrist independently. Next, the end pose of robot $T_{2,j} (j = 1, 2, \dots, n)$ is measured at each configuration, as shown in Figure 3c.

The composition is conducted according to Equation (4) so that we can obtain the end poses of the hybrid robot T_k at a large number of combined configurations $q_k = (u_i, v_j) = (q_{1,i}, q_{2,i}, q_{3,i}, q_{4,j}, q_{5,j}, q_{6,j})$:

$$T_k = T_{1,i} \cdot T_0^{-1} \cdot T_{2,j}, k = 1, 2, \dots, m \times n \quad (5)$$

The six-dimensional joint space of the hybrid robot is decomposed into two three-dimensional subspaces through decomposition measurement. Assuming that m and n measurement configurations are planned in two subspaces, the robot poses at $m \times n$ configurations can be easily obtained by only $m + n$ measurements, which greatly improves the measurement efficiency. Moreover, the method is superior in terms of measurement efficiency with the increase of the measurement configurations m and n .

4. Improved Data-Driven Methodology for Calibration

In this section, an improved data-driven methodology is proposed. It directly establishes the mapping between the nominal joint variables and joint compensation values based on a BPNN and then conducts the training by a unique algorithm involving the robot inverse kinematics.

4.1. Principle of the Improved Data-Driven Methodology

The existing data-driven methods usually include two steps: estimation and compensation. First, an error estimation model is established based on the data-driven approach. Next, compensation is implemented based on the estimated values, as described in our previous research [40]. Although the method realized real-time compensation and achieved a high calibration accuracy in the whole workspace of the robot, it inevitably introduces the estimated residual in theory. Moreover, the joint compensation values are calculated based on Jacobian iteration, which may cause a heavy computational burden in the sample set construction. Hence, an improved data-driven methodology is proposed. By merging the processes of estimation and compensation, a mapping between the nominal joint variables and joint compensation values is established based on a BPNN and then trained directly using the measurement data through a unique training algorithm involving the robot inverse kinematics.

As shown in Figure 4, the data-driven methodology mainly contains two steps: offline calibration and online compensation. First, measurement is conducted to obtain the actual pose y_m of the hybrid robot at the nominal joint variable q_n , and the joint compensation value Δq is computed according to the robot mechanism model such as the kinematic and Jacobian models. Second, the mapping $g(q_n, \alpha)$ between the nominal joint variables and joint compensation values is established based on the data-driven methodology, where $g(q_n, \alpha)$ can be a polynomial function, interpolation function, or neural network, and α represents the parameter to be determined by the mapping. Next, parameter $\hat{\alpha}$ is fitted to minimize the predicted residuals for all measurement configurations. Finally, the compensator $g(q_n, \hat{\alpha})$ is embedded into the robot control system for online compensation. For a certain joint command q_n , the joint compensation value $\Delta \hat{q}$ is obtained according to the compensator, and the corrected joint variable q_a is calculated by $q_n - \Delta \hat{q}$ so that the robot's actual pose y_a can be as close as possible to the desired pose y_n .

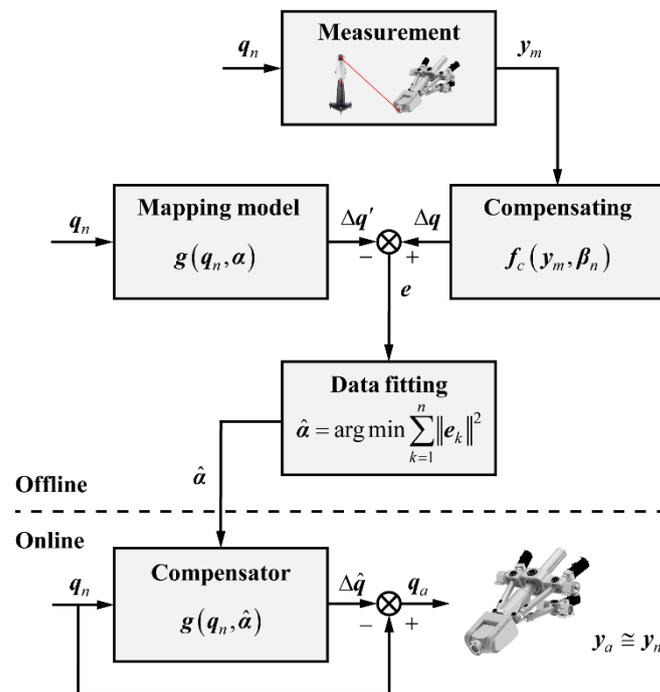


Figure 4. The scheme of the improved data-driven methodology.

The method establishes the mapping model between the nominal joint variables and joint compensation values, which can be directly used for online compensation after data fitting, voiding the troubles of error estimation and iterative calculation in compensation. To guarantee the implementation effect, the following two core problems need to be solved: (1) how to establish an accurate mapping model; and (2) how to obtain accurate joint compensation value.

4.2. Mapping Model Designing

The first step is to establish an appropriate mapping model. According to previous research, polynomial fitting can accurately estimate the position errors of robots on Cartesian space trajectory [22]. However, for the compensation of pose errors in the whole workspace of a 6-DOF robot, constructing high-order polynomials with six variables will involve too many coefficient terms, which will cause much trouble in parameter identification [23]. Furthermore, building up a query table of spatial interpolation in the whole workspace of the robot requires the expensive cost of time and storage space [9], so it is tough to implement in practice. The neural network, which has a strong ability for nonlinear mapping and generalization, can fit any complex nonlinear mapping. In recent years, a back propagation neural network (BPNN) has been gradually applied in robot calibration by many scholars [28,29]. Along this track, a BPNN is built to describe the mapping between the nominal joint variables and joint compensation values in the whole workspace of the robot.

Taking the nominal joint variable and corresponding compensation value as the input and output, respectively, a BPNN is constructed, as illustrated in Figure 5. Both the input and output layers consist of six neurons, representing six elements of the nominal joint variable q_n and the joint compensation value Δq , respectively. According to some successful experiences [31], it has two hidden layers, and the tan-sigmoid function and linear function are taken as the activation functions of the hidden layer and the output layer, respectively.

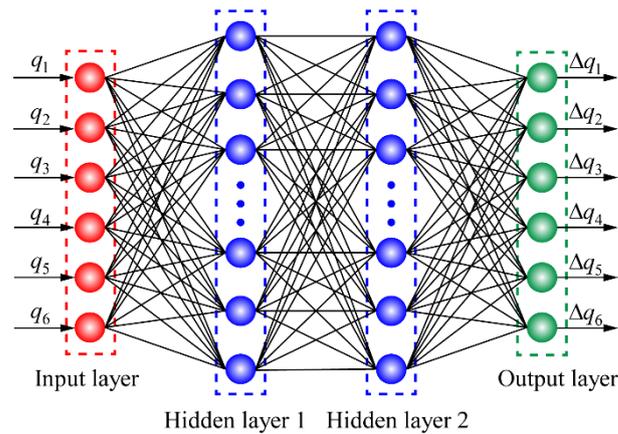


Figure 5. The structure of the BPNN.

4.3. Network Training

Next, a unique network-training algorithm is proposed based on the robot inverse kinematics, which can train the network directly using the pre-measured robot poses, avoiding the troubles of calculating the pose errors and solving the joint compensation values. The principle of network training is shown in Figure 6.

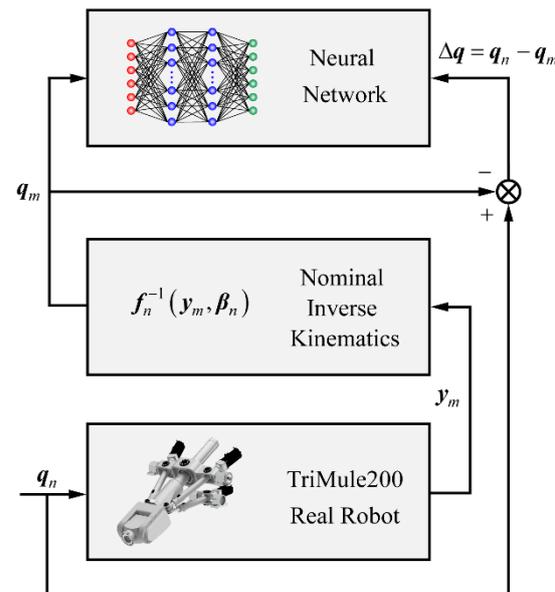


Figure 6. The principle of network training.

When the nominal joint variable q_n is input, the robot moves to y_m rather than the desired pose y_n . Next, the nominal joint variable q_m of pose y_m is computed through the nominal inverse kinematics. Assuming that the desired pose of the robot is y_m , the joint command q_m theoretically needs to be input to the robot. However, the robot cannot move to y_m due to various errors in the robot system. The robot will move to y_m exactly when the joint command q_n is input at the beginning. Hence, the joint variable q_n can be regarded as the corrected joint variable of the desired pose y_m , and $\Delta q = q_n - q_m$ is the corresponding joint compensation value of robot configuration $y_m(q_m)$. It is worth noting that Δq here is the joint compensation value corresponding to joint variable q_m rather than q_n . Therefore, the neural network should be trained with q_m and Δq as the input and output, respectively.

Following the principle above, Algorithm 1 is proposed for training the BPNN:

Algorithm 1: Training of BPNN

-
- 1 **foreach** $q_n \in \text{Training_set}$ **do**;
 - 2 Move to q_n and measure the actual pose y_m ;
 - 3 Calculate the nominal joint variable q_m of y_m by $f_n^{-1}(y_m, \beta_n)$;
 - 4 Calculate the joint compensation value $\Delta q = q_n - q_m$;
 - 5 Train the BPNN with q_m as input and Δq as output.
-

Through the algorithm, the BPNN can be trained directly based on the pre-measured robot poses and the nominal inverse kinematics, which avoids the process of solving robot pose errors and the complex iterative calculation in error compensation. In terms of compensation accuracy, the real compensation value of the robot can be obtained on the premise of ignoring the measurement errors rather than the approximate value obtained by iterative calculation.

5. Experiments

To demonstrate the effectiveness of the methodology, the validation experiments are carried out as shown in Figure 7. A TriMule 200 robot, which has a repeatability accuracy of 0.0197 mm/0.0041 deg, is taken as the verification platform. The measuring instrument is a Leica AT901-LR laser tracker that has a maximal observed deviation of 0.005 mm/m. In addition, all measurements are conducted under static and no-load conditions.

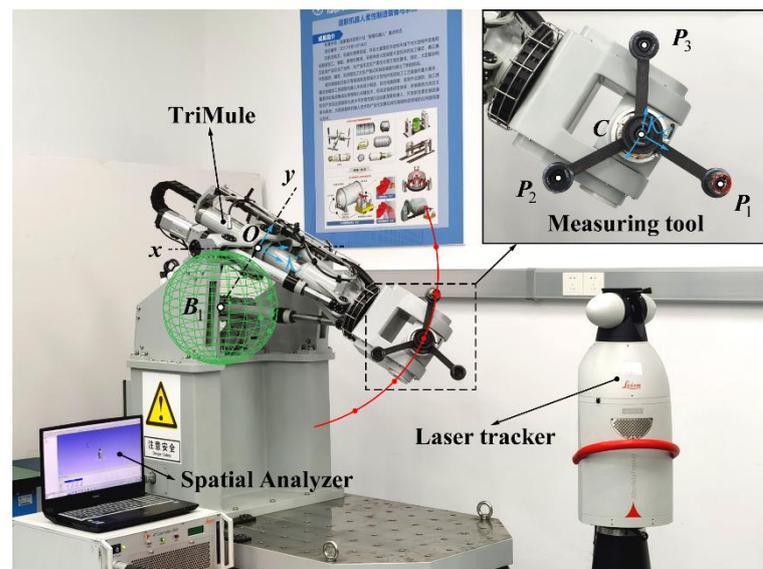


Figure 7. Experimental setup for the calibration of the TriMule robot.

Before pose measurement, the base frame \mathcal{K} is registered in the measuring software (Spatial Analyzer) according to its definition, as illustrated in Section 2. The x -axis is constructed by fitting the arc trajectory marked in red, which is formed by the spherically mounted retroreflector (SMR) attached to the end of the robot. The center B_1 of the U joint is constructed by fitting the green spherical surface, which is formed by another SMR fixed at limb 1. A normal line from B_1 to the x -axis is constructed as the y -axis, and intersection O is defined as the origin. Next, a base frame \mathcal{K} is constructed according to the right hand rule (see Figure 7). A specialized measuring tool is connected to the end effector with its centers of the three SMRs defined as P_1 , P_2 , and P_3 . The end frame \mathcal{K}_C is placed at point C , which is the center of P_1 , P_2 , and P_3 . Thus, the end pose T_C can be obtained by measuring P_1 , P_2 , and P_3 as follows:

$$T_C = \begin{bmatrix} R_C & r_C \\ 0 & 1 \end{bmatrix} \quad (6)$$

with

$$\mathbf{R}_C = [\mathbf{n}_x \quad \mathbf{n}_y \quad \mathbf{n}_z] \in \mathbb{R}^{3 \times 3}, \quad \mathbf{r}_C = \frac{P_1 + P_2 + P_3}{3}$$

$$\mathbf{n}_x = \frac{(P_2 - P_3)}{\|(P_2 - P_3)\|_2}, \quad \mathbf{n}_z = \frac{(P_2 - P_3) \times (P_1 - P_2)}{\|(P_2 - P_3) \times (P_1 - P_2)\|_2}, \quad \mathbf{n}_y = \mathbf{n}_z \times \mathbf{n}_x$$

where \mathbf{n}_x is aligned with $\vec{P_3 P_2}$, and \mathbf{n}_z is normal to $\Delta P_1 P_2 P_3$.

The decomposition measurement methodology is adopted to efficiently acquire the robot poses in the whole workspace. The joint space of the hybrid robot is divided into two subspaces, i.e., the joint space of the parallel mechanism (W1) composed of joints 1, 2, and 3 and the joint space of the wrist (W2) consisting of joints 4, 5, and 6. The joint space of the hybrid robot is defined by the motion ranges of the six actuated joints, as shown in Table 1. To sample the robot poses over the entire workspace, W1 and W2 are discretized into two sets of three-dimensional grid elements, of which vertices are taken as the sampling configurations of W1 and W2, respectively. The discretization rules are as follows: (1) Joints 1, 2, and 3 are divided by a sampling interval of 60 mm so that W1 is discretized into a group of 125 sampling configurations. (2) The sampling intervals of 60° , 22.5° , and 45° are adopted for joints 4, 5, and 6, respectively, to discretize W2 into a group of 175 sampling configurations.

Table 1. Definition of the robot joint space.

Joint	Range (mm/ $^\circ$)
1	[−70, 170]
2	[−20, 220]
3	[−20, 220]
4	[−180, 180]
5	[0, 90]
6	[−90, 90]

The home pose of the robot is taken as the reference configuration for convenience. Following the detailed processes described in Section 3.2, the end poses of the robot are measured at the sampling configurations of W1 and W2, as illustrated in Figure 8a,b. Figure 8c shows the distribution diagram of the tool center point (TCP) in the robot workspace, which is obtained by composing the sampling configurations of W1 and W2. Consequently, a group of 21,875 samples are acquired with only 300 measurements.

To prove the effectiveness of the proposed method, it is compared with the calibration method in reference [38]. For more detailed steps of the comparative experiment, please refer to [38], which will not be repeated here. This paper only gives the necessary measurement process, network training parameters, and error compensation results.

Both methods adopt the same sampling configurations as described before. The detailed measurement procedures and sampling time are shown in Tables 2 and 3, respectively. The experimental results show that both methods effectively improve the measurement efficiency, which can obtain massive sampling data in the whole workspace of the robot within 160 min. Although the proposed method does not show significant advantages in sampling time, it avoids the establishment of the transfer frame and frequent coordinate system transformation in the measurement. All data are directly measured based on the robot base frame and the end measuring tool. Thus, it is more practical and efficient for on-site implementation.

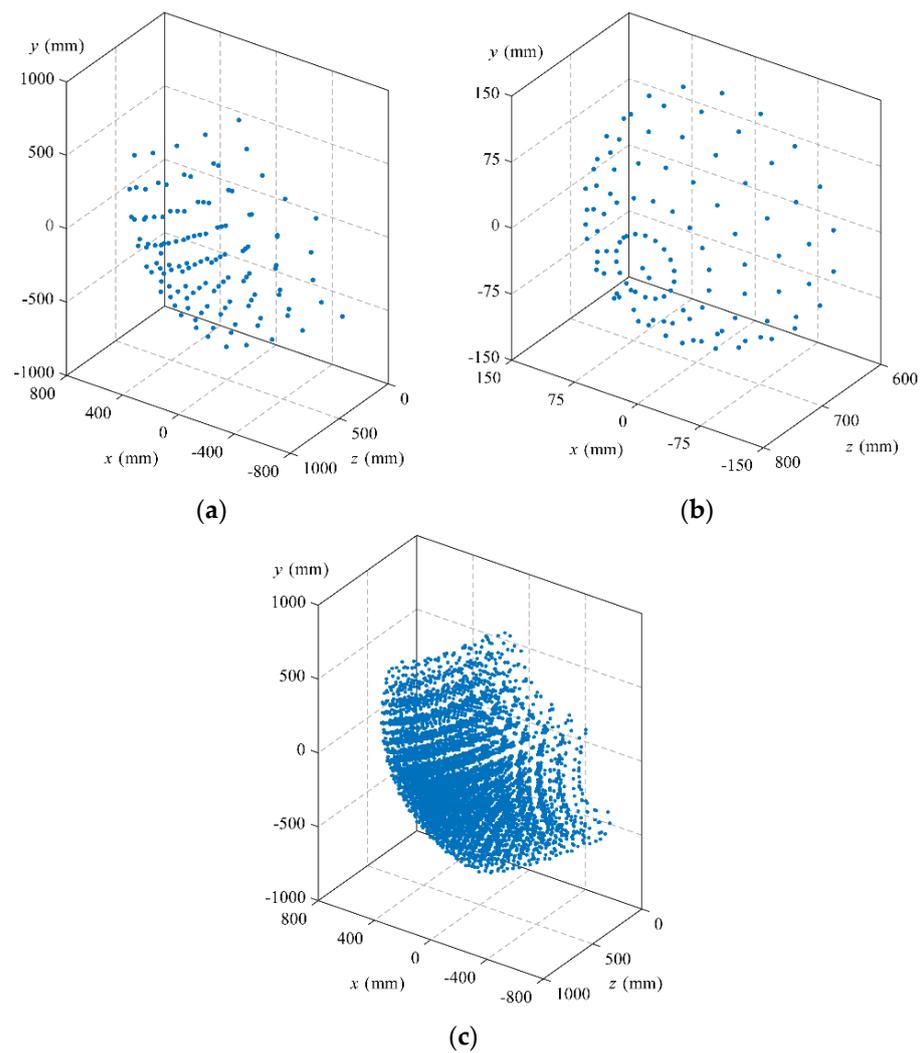


Figure 8. Distribution diagram of the TCP in the robot workspace: (a) Sampling configurations of W1; (b) Sampling configurations of W2; and (c) Configurations through composition.

Table 2. Sampling time of the proposed method.

Measurement Procedure	Sampling Time (min)
Establishment of base frame	20
Measurement of the reference configuration	1
Measurement by driving parallel mechanism (125 configurations)	50
Measurement by driving wrist (175 configurations)	70
Total measurement process	141

Table 3. Sampling time of the comparative method.

Measurement Procedure	Sampling Time (min)
Establishment of base frame	20
Establishment of transfer frame	20
Measurement of parallel mechanism (125 configurations)	50
Measurement of wrist (175 configurations)	70
Total measurement process	160

Next, two BPNNs are trained by two methods, respectively, to compensate for the robot pose errors. Since the hyperparameters of the network will directly affect its training

performance, a series of comparative experiments are conducted to determine the optimal number of hidden layer neurons. The number of neurons is increased gradually to search and verify the optimal architecture, which is judged by the root mean square error (RMSE) of the testing and training sets [21,31]. Table 4 shows the specific training parameters of two neural networks. It is worth noting that the networks are trained by two different sample sets constructed by the proposed and comparative methods, respectively, although they have the same number of samples. Finally, the BPNNs are embedded into the robot to realize online compensation.

Table 4. Training parameters of two neural networks.

Parameters	Proposed Method	Comparative Method
Training samples	21,875	21,875
Hidden neurons	30, 15	28, 16
Initialization	Xavier	Xavier
Training function	LM	LM
Training time (s)	135	147
Training RMSE (mm)	0.0367	0.0384
Testing RMSE (mm)	0.0425	0.0419

The uniform sampling intervals of 80 mm, 80 mm, 80 mm, 90°, 30°, and 60° are adopted for joints 1, 2, 3, 4, 5, and 6 to discretize the whole workspace of the hybrid robot into 5120 poses, from which 100 poses are randomly selected as validation configurations, as shown in Figure 9. It is worth mentioning that none of the discretization poses overlap the sampling poses. The actual pose of the robot is measured before and after compensation and then compared with its nominal pose. The volumetric orientation error $\Delta\theta$ and volumetric position error Δr (simplified as the orientation and position errors) are taken as the evaluation index of robot accuracy.

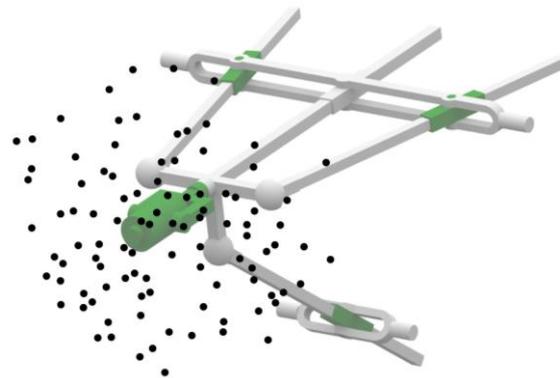


Figure 9. Distribution of the validation configurations.

Figures 10 and 11 and Table 5 show the experimental results. After compensation by the proposed method, the robot's maximum position/orientation errors are reduced to 0.085 mm/0.022 deg, which is 91.16%/88.17% lower than 0.962 mm/0.186 deg before compensation, respectively, and the mean position/orientation errors are reduced by 91.22%/89.74% to 0.049 mm/0.012 deg, respectively. After compensation by the comparative method, the maximum position/orientation errors have decreased to 0.098 mm/0.024 deg, which is 89.81%/86.56% lower than before compensation, and the mean position/orientation errors are reduced by 89.96%/88.73% to 0.056 mm/0.014 deg, respectively. Thus, we can conclude that the methodology can effectively improve the pose accuracy in the whole workspace of the robot.

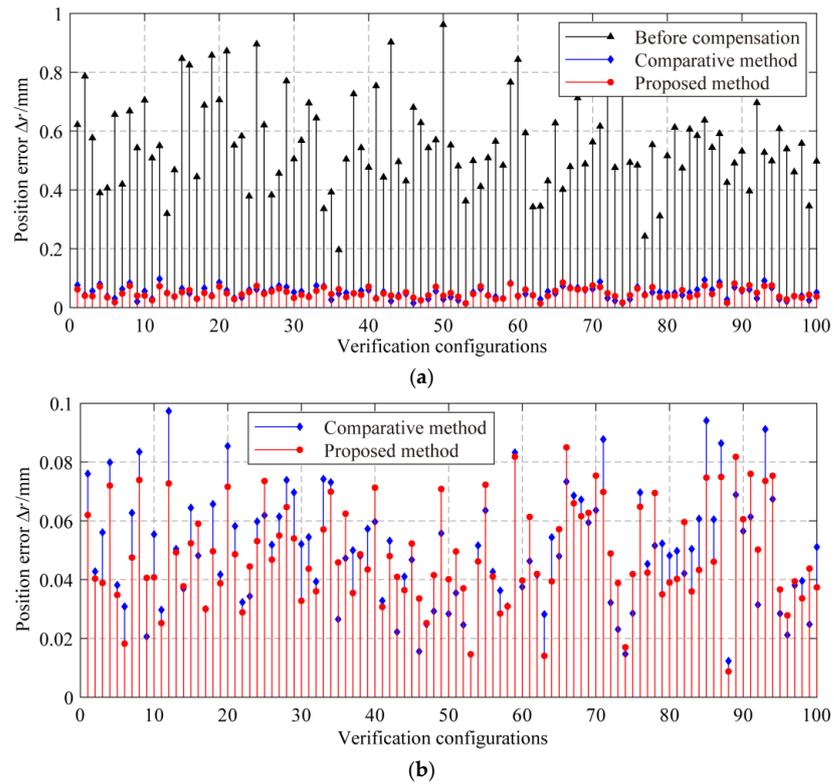


Figure 10. The position errors before and after compensation: (a) The position errors; and (b) The magnified graph of (a), the residual position errors after compensation by two methods.

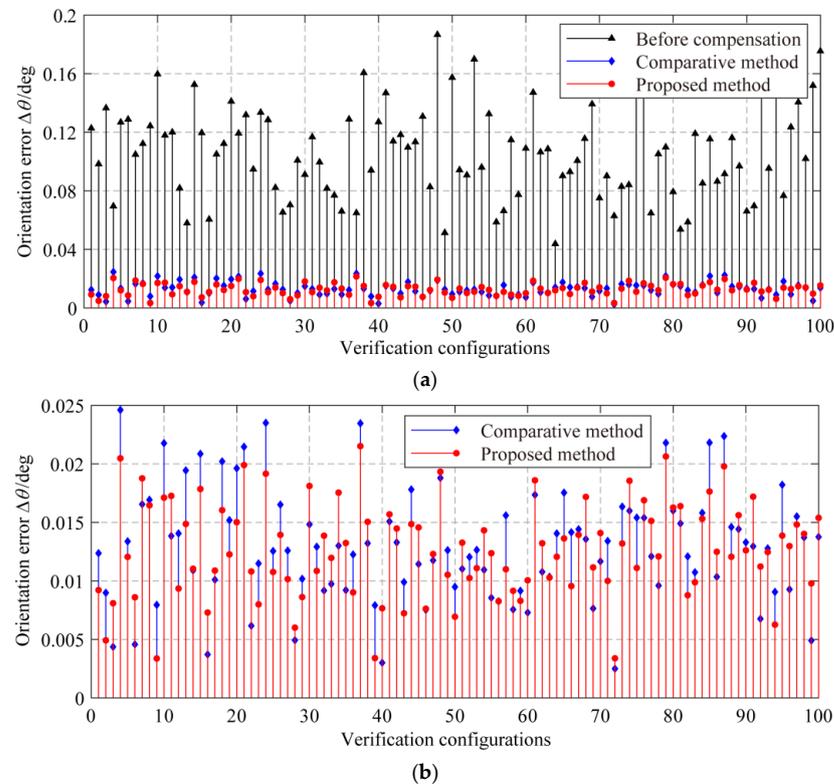


Figure 11. The orientation errors before and after compensation: (a) The orientation errors; and (b) The magnified graph of (a), the residual orientation errors after compensation by two methods.

Table 5. Pose errors before and after compensation.

Pose Errors		Max	Mean	Rms
Before compensation	Δr (mm)	0.962	0.558	0.579
	$\Delta\theta$ (deg)	0.186	0.117	0.122
Proposed method	Δr (mm)	0.085	0.049	0.052
	$\Delta\theta$ (deg)	0.022	0.012	0.013
Comparative method	Δr (mm)	0.098	0.056	0.054
	$\Delta\theta$ (deg)	0.024	0.014	0.015

6. Conclusions

This paper proposes an improved data-driven calibration method for a 6-DOF hybrid robot. It mainly focuses on enhancing calibration efficiency and practicability. The following conclusions are drawn.

(1) A decomposition measurement method is proposed to overcome the curse of dimensionality in the measurement of data-driven calibration. An arbitrary motion of the hybrid robot is equivalently decomposed into three independent sub-motions through motion decomposition. The sub-motions are sequentially combined according to specific motion rules. Thus, a large number of robot poses can be acquired over the entire workspace with a limited number of measurements.

(2) An improved data-driven methodology is proposed to replace the traditional processes of error estimation and compensation to improve the practicability in field applications. The mapping between the nominal joint variables and joint compensation values is established based on a BPNN. Next, it is trained directly using the measurement data of robot poses through a unique training algorithm involving the robot inverse kinematics.

(3) The experimental results demonstrate the effectiveness of the method. The robot's maximal position/orientation errors are reduced by 91.16%/88.17% to 0.085 mm/0.022 deg after calibration.

(4) The proposed method can also be applied to other serial or hybrid robots as a general methodology.

In the future, we will continue to refine this method. In measurement, the relationship between the distributions of sampling configurations and calibration accuracy has to be further investigated to provide guidance for sampling configuration optimization. In network training, the research on an intelligent optimization algorithm of network structure will be carried out to replace the complex comparative experiments at present.

Author Contributions: Conceptualization, Z.Y.; Methodology, Z.Y., H.L. and T.H.; Software, Y.W.; Formal analysis, Z.Y. and J.X.; Investigation, Z.Y.; Resources, Y.W. and T.H.; Data curation, Z.Y. and J.X.; Writing—original draft, Z.Y.; Writing—review & editing, H.L.; Visualization, Z.Y.; Supervision, T.H.; Project administration, Y.W., H.L. and J.X.; Funding acquisition, H.L., J.X. and T.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. 91948301 and 51721003).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Luo, X.; Xie, F.; Liu, X.; Xie, Z. Kinematic calibration of a 5-axis parallel machining robot based on dimensionless error mapping matrix. *Robot. Comput. Integr. Manuf.* **2021**, *70*, 102115. [[CrossRef](#)]
- Hernández-Martínez, E.; López-Cajún, C.; Jáuregui-Correa, J. Calibration of parallel manipulators and their application to machine tools. *Ing. Investig. Tecnol.* **2010**, *11*, 141–154.

3. Wu, Y.; Klimchik, A.; Caro, S.; Furet, B.; Pashkevich, A. Geometric calibration of industrial robots using enhanced partial pose measurements and design of experiments. *Comput. Integr. Manuf.* **2015**, *35*, 151–168. [[CrossRef](#)]
4. Dong, C.; Liu, H.; Yue, W.; Huang, T. Stiffness modeling and analysis of a novel 5-DOF hybrid robot. *Mech. Mach. Theory* **2018**, *125*, 80–93. [[CrossRef](#)]
5. Idá, E.; Merlet, J.-P.; Carricato, M. Automatic self-calibration of suspended under-actuated cable-driven parallel robot using incremental measurements. *Cable-Driven Parallel Robots* **2019**, *74*, 333–344.
6. Daney, D.; Emiris, I.Z.; Papegay, Y.; Tsigaridas, E.; Merlet, J.-P. Calibration of parallel robots: On the Elimination of Pose-Dependent Parameters. In Proceedings of the 1st European Conference on Mechanism Science, Obergurgl, Austria, 21–26 February 2006.
7. Legnani, G.; Tosi, D.; Adamini, R. Calibration of Parallel Kinematic Machines: Theory and Applications. *Ind. Robot.* **2004**, 171.
8. Huang, T.; Zhao, D.; Yin, F.; Tian, W.; Chetwynd, D.G. Kinematic calibration of a 6-DOF hybrid robot by considering multicollinearity in the identification Jacobian. *Mech. Mach. Theory* **2019**, *131*, 371–384. [[CrossRef](#)]
9. Chen, G.; Li, T.; Chu, M.; Xuan, J.; Xu, S. Review on kinematics calibration technology of serial robots. *Int. J. Precis. Eng. Manuf.* **2014**, *15*, 1759–1774. [[CrossRef](#)]
10. Zhang, D.; Gao, Z. Optimal kinematic calibration of parallel manipulators with pseudoerror theory and cooperative coevolutionary network. *IEEE Trans. Ind. Electron.* **2012**, *59*, 3221–3231. [[CrossRef](#)]
11. Cao, S.; Cheng, Q.; Guo, Y.; Zhu, W.; Wang, H.; Ke, Y. Pose error compensation based on joint space division for 6-DOF robot manipulators. *Precis. Eng.-J. Int. Soc. Precis. Eng. Nanotechnol.* **2022**, *74*, 195–204. [[CrossRef](#)]
12. Ma, L.; Bazzoli, P.; Sammons, P.M.; Landers, R.G.; Bristow, D.A. Modeling and calibration of high-order joint-dependent kinematic errors for industrial robots. *Robot. Comput. Integr. Manuf.* **2018**, *50*, 153–167. [[CrossRef](#)]
13. Shen, N.; Yuan, H.; Li, J.; Wang, Z.; Geng, L.; Shi, H.; Lu, N. Efficient model-free calibration of a 5-Degree of freedom hybrid robot. *J. Mech. Robot.* **2022**, *14*, 051011. [[CrossRef](#)]
14. Shamma, J.S.; Whitney, D.E. A Method for Inverse Robot Calibration. *J. Dyn. Syst. Meas. Control Trans. ASME* **1987**, *109*, 36–43. [[CrossRef](#)]
15. Bai, Y.; Wang, D. On the comparison of trilinear, cubic spline, and fuzzy interpolation methods in the high-accuracy measurements. *IEEE Trans. Fuzzy Syst.* **2010**, *18*, 1016–1022.
16. Chen, D.; Yuan, P.; Wang, T.; Cai, Y.; Xue, L. A compensation method for enhancing aviation drilling robot accuracy based on co-kriging. *Int. J. Precis. Eng. Manuf.* **2018**, *19*, 1133–1142. [[CrossRef](#)]
17. Tian, W.; Mei, D.; Li, P.; Zeng, Y.; Zhou, W. Determination of optimal samples for robot calibration based on error similarity. *Chin. J. Aeronaut.* **2015**, *28*, 946–953. [[CrossRef](#)]
18. Liao, S.; Zeng, Q.; Ehmman, K.F.; Cao, J. Parameter identification and nonparametric calibration of the Tri-pyramid robot. *IEEE-ASME Trans. Mechatron.* **2020**, *25*, 2309–2317. [[CrossRef](#)]
19. Jang, J.H.; Kim, S.H.; Kwak, Y.K. Calibration of geometric and non-geometric errors of an industrial robot. *Robotica* **2001**, *19*, 311–321. [[CrossRef](#)]
20. Nguyen, H.N.; Zhou, J.; Kang, H.J. A calibration method for enhancing robot accuracy through integration of an extended Kalman filter algorithm and an artificial neural network. *Neurocomputing* **2015**, *151*, 996–1005. [[CrossRef](#)]
21. Angelidis, A.; Vosniakos, G.C. Prediction and compensation of relative position error along industrial robot end-effector paths. *Int. J. Precis. Eng. Manuf.* **2014**, *15*, 63–73. [[CrossRef](#)]
22. Alici, G.; 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. [[CrossRef](#)]
23. Alici, G.; Jagielski, R.; Şekercioglu, Y.A.; Shirinzadeh, B. Prediction of geometric errors of robot manipulators with Particle Swarm Optimisation method. *Robot. Auton. Syst.* **2006**, *54*, 956–966. [[CrossRef](#)]
24. Bai, Y.; Wang, D. Calibrate parallel machine tools by using interval type-2 fuzzy interpolation method. *Int. J. Adv. Manuf. Technol.* **2017**, *93*, 3777–3787. [[CrossRef](#)]
25. Tian, W.; Zeng, Y.; Zhou, W.; Liao, W. Calibration of robotic drilling systems with a moving rail. *Chin. J. Aeronaut.* **2014**, *27*, 1598–1604. [[CrossRef](#)]
26. Cai, Y.; Yuan, P.; Shi, Z.; Chen, D.; Cao, S. Application of universal Kriging for calibrating offline-programming industrial robots. *J. Intell. Robot. Syst.* **2019**, *94*, 339–348. [[CrossRef](#)]
27. Chen, D.; Wang, T.; Yuan, P.; Sun, N.; Tang, H. A positional error compensation method for industrial robots combining error similarity and radial basis function neural network. *Meas. Sci. Technol.* **2019**, *30*, 125010. [[CrossRef](#)]
28. Wang, D.; Bai, Y.; Zhao, J. Robot manipulator calibration using neural network and a camera based measurement system. *Trans. Inst. Meas. Control* **2010**, *34*, 105–121. [[CrossRef](#)]
29. Zhang, D.; Zhang, G.; Li, L. Calibration of a six-axis parallel manipulator based on BP neural network. *Ind. Robot.* **2019**, *46*, 692–698. [[CrossRef](#)]
30. Yuan, P.; Chen, D.; Wang, T.; Cao, S.; Cai, Y. A compensation method based on extreme learning machine to enhance absolute position accuracy for aviation drilling robot. *Adv. Mech. Eng.* **2018**, *10*, 1687814018763411. [[CrossRef](#)]
31. Zhao, G.; Zhang, P.; Ma, G.; Xiao, W. System identification of the nonlinear residual errors of an industrial robot using massive measurements. *Robot. Comput. Integr. Manuf.* **2019**, *59*, 104–114. [[CrossRef](#)]
32. Ulbrich, S.; De Angulo, V.R.; Asfour, T.; Torras, C.; Dillmann, R. General robot kinematics decomposition without intermediate markers. *IEEE Trans. Neural Netw. Learn. Syst.* **2012**, *23*, 620–630. [[CrossRef](#)] [[PubMed](#)]

33. Belchior, J.; Guillo, M.; Courteille, E.; Maurine, P.; Leotoing, L.; Guines, D. Off-line compensation of the tool path deviations on robotic machining: Application to incremental sheet forming. *Robot. Comput. Integr. Manuf.* **2013**, *29*, 58–69. [[CrossRef](#)]
34. Zeng, Y.; Tian, W.; Liao, W. Positional error similarity analysis for error compensation of industrial robots. *Robot. Comput. Integr. Manuf.* **2016**, *42*, 113–120. [[CrossRef](#)]
35. Chen, D.; Yuan, P.; Wang, T.; Cai, Y.; Tang, H. A compensation method based on error similarity and error correlation to enhance the position accuracy of an aviation drilling robot. *Meas. Sci. Technol.* **2018**, *29*, 085011. [[CrossRef](#)]
36. Guo, Y.; Yin, S.; Ren, Y.; Zhu, J.; Yang, S.; Ye, S. A multilevel calibration technique for an industrial robot with parallelogram mechanism. *Precis. Eng. J. Int. Soc. Precis. Eng. Nanotechnol.* **2015**, *40*, 261–272. [[CrossRef](#)]
37. Xiong, G.; Ding, Y.; Zhu, L.; Su, C. A product-of-exponential-based robot calibration method with optimal measurement configurations. *Int. J. Adv. Robot. Syst.* **2017**, *14*, 1729881417743555. [[CrossRef](#)]
38. Xie, P.; Du, Y.; Tian, P.; Liu, B. A parallel robot error comprehensive compensation method. *Chin. J. Mech. Eng.* **2012**, *48*, 43–49. [[CrossRef](#)]
39. Dolinsky, J.U.; Jenkinson, I.D.; Colquhoun, G.J. Application of genetic programming to the calibration of industrial robots. *Comput. Ind.* **2007**, *58*, 255–264. [[CrossRef](#)]
40. Liu, H.; Yan, Z.; Xiao, J. Pose error prediction and real-time compensation of a 5-DOF hybrid robot. *Mech. Mach. Theory* **2022**, *170*, 104737. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.