



# An Enhanced Positional Error Compensation Method for Rock Drilling Robots Based on LightGBM and RBFN

Xuanyi Zhou<sup>1</sup>, Wenyu Bai<sup>1,2\*</sup>, Jilin He<sup>3</sup>, Ju Dai<sup>3</sup>, Peng Liu<sup>4</sup>, Yuming Zhao<sup>4,5</sup> and Guanjun Bao<sup>1</sup>

<sup>1</sup> Key Laboratory of Special Purpose Equipment and Advanced Processing Technology, Zhejiang University of Technology, Hangzhou, China, <sup>2</sup> Zhejiang Jinbang Sports Equipment Co. Ltd., Zhejiang, China, <sup>3</sup> State Key Laboratory of High Performance Complex Manufacturing, Central South University, Changsha, China, <sup>4</sup> Sunward Intelligent Equipment Company, Ltd., Changsha, China, <sup>5</sup> Department of Precision Instruments, Tsinghua University, Beijing, China

Rock drilling robots are able to greatly reduce labor intensity and improve efficiency and quality in tunnel construction. However, due to the characteristics of the heavy load, large span, and multi-joints of the robot manipulator, the errors are diverse and non-linear, which pose challenges to the intelligent and high-precision control of the robot manipulator. In order to enhance the control accuracy, a hybrid positional error compensation method based on Radial Basis Function Network (RBFN) and Light Gradient Boosting Decision Tree (LightGBM) is proposed for the rock drilling robot. Firstly, the kinematics model of the robotic manipulator is established by applying MDH. Then a parallel difference algorithm is designed to modify the kinematics parameters to compensate for the geometric error. Afterward, non-geometric errors are analyzed and compensated by applying RBFN and lightGBM including features and kinematics model. Finally, the experiments of the error compensation by combining combining the geometric and non-geometric errors verify the performance of the proposed method.

**Keywords:** rock drilling robot, error compensation, parameter identification, parallel differential evolution algorithm, RBFN

## OPEN ACCESS

### Edited by:

Hang Su,  
Fondazione Politecnico di Milano, Italy

### Reviewed by:

Jing Luo,  
Wuhan Institute of Technology, China  
Peiyuan Guan,  
University of Oslo, Norway  
Xiaoshan Qian,  
Yichun University, China

### \*Correspondence:

Wenyu Bai  
bauyo@zjut.edu.cn

**Received:** 25 February 2022

**Accepted:** 14 March 2022

**Published:** 13 May 2022

### Citation:

Zhou X, Bai W, He J, Dai J, Liu P, Zhao Y and Bao G (2022) An Enhanced Positional Error Compensation Method for Rock Drilling Robots Based on LightGBM and RBFN. *Front. Neurobot.* 16:883816. doi: 10.3389/fnbot.2022.883816

## 1. INTRODUCTION

The tunnel construction machine has been playing an increasingly important role in the development of the modern economy due to the continuous expansions of anthropogenic activity. Moreover, the drilling and blasting method is the most practical and effective method for tunnel engineering (Ocak and Bilgin, 2010). Rock drilling robots are one of an essential piece of tunnel equipment used for drilling and blasting methods. They have been widely applied in the tunnel construction due to its their advantages for reducing labor intensity. Robotic drilling can effectively prevent overcutting and undercutting to improve the efficiency and quality of the tunnel. However, due to the characteristics of the heavy load, large span, and multi-joints of the robotic manipulator, the errors of the rock drilling robot are diverse and non-linear, which pose challenges to the intelligent and high-precision control of the robotic manipulator.

Position accuracy is an important parameter for the robot (Chen D. et al., 2018; Qi et al., 2021). High positioning accuracy is the a guarantee for complex robotic tasks. Especially for

industrial robots and medical robots, the error compensation method is important for efficiency and safety. Now robotic surgery is gradually replacing manual surgery because of its stability and convenience, and high precision is the most important guarantee element needed for safety (Su et al., 2018, 2021). These electrical robots for the industry can achieve high accuracy of up to a millimeter level with a high-precision motor (Su et al., 2022). However, the accuracy of hydraulic robots is limited by the underlying rigid actuation mechanisms of the hydraulic actuators. With the improvement of accuracy, the price of hydraulic components has also increased substantially. Therefore, it is necessary to use error compensation methods for the hydraulic robot utilizing conventional hydraulic actuators.

In general, considering the characteristics of nonlinear and multi-coupling, the robotic error is very complex. With the advancement of computer technology, numerous non-linear fitting algorithms are proposed, such as Artificial Neural Networks (ANN) (Zhang et al., 2021), Radial Basis Function Network (RBFN) (Park and Sandberg, 1991), and extreme learning machine (ELM) (Huang et al., 2004), etc. Huang proposed a method to assess the critical parameters for short-term wind generation forecasting by ANN (Sewdien et al., 2020). Chen D. et al. (2019) proposed a positional error compensation combining RBFN and error similarity. RBFN is used to estimate the error of the target positions. Yuan et al. (2018) demonstrated that a trained ELM method could guarantee high position accuracy on the drilling robot and reduce the working time and workload. Neural networks have a very wide application. However, it needs many super-parameters to set and takes too long to train the process.

Decision Tree (DT) is a supervised learning method that is widely used to predict models based on a tree structure. DT has higher decision-making efficiency in comparison with other machine learning methods. In fact, many researchers have improved DT models, and some more optimized algorithms have been proposed, such as Random Forest (RF) (Breiman, 2001; Svetnik et al., 2003), Gradient Boosting Decision Tree (GBDT) (Ke et al., 2017), eXtreme Gradient Boosting (XGBoost) (Naghibi et al., 2020), and so on. The liquid crystalline behaviors are successfully predicted using RF in Chen C. H. et al. (2018). Yao et al. (2019) proposed a method for predicting line loss rate in a low voltage distribution network based on GBDT method. By analyzing and verifying the data, the GBDT method is accurate and effective. Light-GBM is also an improved algorithm based on GBDT, which was firstly proposed by Microsoft Research Asia in 2016 (Chen P. et al., 2019). Zhou et al. (2020) proposed a hybrid reservoir permeability prediction method based on Light-GBM. The results show that the method has excellent prediction ability. Compared with the DT algorithm, Light-GBM can significantly promote the training speed without decreasing the accuracy and also occupy less memory during the training process.

There are two types of positional errors in robotic control. The first error is a geometric error, which is generated by the mechanical manufacturing error and assembling error of the robot. To reduce the geometric errors, a parallel difference algorithm is designed in this article to modify the kinematics parameters (Praveen and Denis, 2018; Zhu et al., 2020). The

second error is generated by environmental factors, such as temperature and gravity, etc., that have nothing to do with the robot itself, which are defined as non-geometric errors (Zeng et al., 2017; Jiang et al., 2021). These non-geometric errors are usually difficult to analyze accurately because of their non-linear characteristics. Specifically for the manipulators as long as 15 meters, the accuracy is affected by many factors such as elastic deflection and hydraulic components. Nevertheless, Atlas Copco, the world-leading rock drilling robot corporation, can achieve an accuracy of up to 100 mm (Molfino et al., 2008).

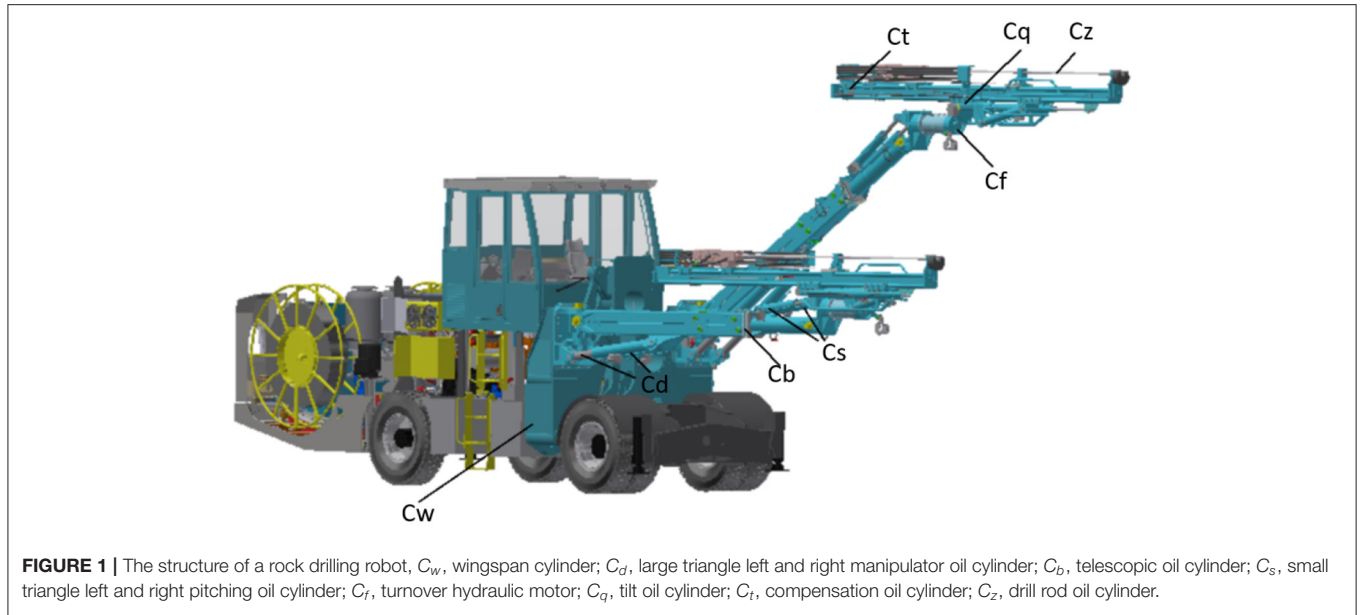
In order to enhance the control accuracy, a hybrid positional error compensation method based on Radial Basis Function Network (RBFN) and Light Gradient Boosting Decision Tree (LightGBM) is proposed for the rock drilling robot. Firstly, the kinematics model of the robotic manipulator is established by applying MDH. Then a parallel difference algorithm is designed to modify the kinematics parameters to compensate for the geometric error. Afterward, non-geometric errors are analyzed and compensated by applying RBFN and lightGBM, including features and the kinematics model. Finally, the experiments of the error compensation by combining the geometric and non-geometric errors verify the performance of the proposed method.

The structure of the article is divided into five parts. The first part is the introduction. The second part is the kinematic model of the rock drilling robot. According to the structural characteristics of the manipulator, the MDH method is applied to establish the kinematics model of the manipulator. The third part is methodology. By analyzing the parameters of the geometric error compensation model, a parallel difference error compensation algorithm is designed to identify the parameters. Moreover, RBFN and lightGBM are applied to compensate for the non-geometric errors. The fourth part is experimental validation and results. The error compensation experiment is carried out, and the effectiveness of the proposed method is verified. The fifth part is the discussion and summary, which discusses the advantages of the error compensation method.

## 2. KINEMATIC MODEL OF THE ROCK DRILLING ROBOT

### 2.1. Mechanical Structure of the Rock Drilling Robot

The rock drilling robot is mainly composed of the wingspan platform, left and right drilling manipulator, as shown in **Figure 1**. The wingspan platform  $C_w$  can be raised to enhance the work area. The drilling manipulator, which is mounted at the end of the wingspan, is the main working device of the rock drilling robot. The drilling manipulator has three moving joints and six rotating joints. The left and right manipulator oil cylinders  $C_d$  are placed at the front end of the manipulator to form a double triangle structure. At the same time, the left and right pitching cylinders  $C_s$  form a small triangle structure and are connected in series with the manipulator front support cylinder  $C_d$  to realize parallel linkage at the rear end of the manipulator. The extension of the manipulator telescopic cylinder  $C_b$  determines



**FIGURE 1 |** The structure of a rock drilling robot,  $C_w$ , wingspan cylinder;  $C_d$ , large triangle left and right manipulator oil cylinder;  $C_b$ , telescopic oil cylinder;  $C_s$ , small triangle left and right pitching oil cylinder;  $C_f$ , turnover hydraulic motor;  $C_q$ , tilt oil cylinder;  $C_t$ , compensation oil cylinder;  $C_z$ , drill rod oil cylinder.

the longitudinal working distance of the manipulator. The hydraulic motor  $C_f$  of the flip joint is used to control the propeller around the axis Linear lateral rotation, with thruster tilt cylinder  $C_q$  to achieve different drilling angles. Compensation cylinder  $C_t$  pushes the bracket to move forward and backward to realize the fine adjustment of the position and posture. Each of the joints is actuated by hydraulic cylinders, which effectively serve as velocity sources.

### 2.2. Forward Kinematics Based on MDH Model

DH method is a standard method to realize the transformation between joint variables and Cartesian coordinates and learn the kinematics modeling of the robot. However, when the joint axes between adjacent links are close to parallel, the position of the actual normal will deviate significantly from the theoretical normal due to the slight deviation of parallelism, which will affect the kinematic calculation.

Therefore, the MDH method is adopted to describe the conversion relationship of parallel joint axis systems. Its characteristic role is to add the rotation change  $\text{Rot}(y_i, \beta_i)$  around the y axis on the original DH model and set the initial value of  $\beta_i$  of adjacent joints to 0.

According to the structural characteristics of the rock drilling manipulator, the coordinate system transformation between links is designed, and the kinematic model of the manipulator is established, as shown in **Figure 2**.

### 2.3. Forward Kinematics Model With MDH

When the coordinate systems of the manipulator 1, 2, 4, and 5 are parallel, the forward kinematics model of the rock drilling robot is established according to MDH method as shown in **Table 1**, and the homogeneous transformation matrix  $T_i^{i-1}$  between adjacent link coordinate systems is obtained, as shown in Equation (1):

$$\begin{aligned}
 T_i^{i-1}(\alpha_{i-1}, a_{i-1}, d_i, \theta_i, \beta_i) &= \text{Trans}(x, a_{i-1}) \text{Rot}(x, \alpha_{i-1}) \\
 &\text{Trans}(z, d_i) \text{Rot}(z, \theta_i) \text{Rot}(y, \beta_i) \\
 &= \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned} \tag{1}$$

where  $m_{11} = c\theta_i c\beta_i$ ,  $m_{12} = -s\theta_i$ ,  $m_{13} = c\theta_i s\beta_i$ ,  $m_{14} = a_{i-1}$ ,  $m_{21} = s\theta_i c\alpha_{i-1} c\beta_i + s\alpha_{i-1} s\beta_i$ ,  $m_{22} = c\theta_i c\alpha_{i-1}$ ,  $m_{23} = s\theta_i c\alpha_{i-1} s\beta_i - s\alpha_{i-1} c\beta_i$ ,  $m_{24} = -d_i s\alpha_{i-1}$ ,  $m_{31} = s\theta_i s\alpha_{i-1} c\beta_i - c\alpha_{i-1} s\beta_i$ ,  $m_{32} = c\theta_i s\alpha_{i-1}$ ,  $m_{33} = s\theta_i s\alpha_{i-1} s\beta_i + c\alpha_{i-1} c\beta_i$ ,  $m_{34} = d_i c\alpha_{i-1}$ .  $s$  presents the sine function while  $c$  is the cosine function.

When the coordinate systems of two adjacent joints are not parallel,  $\beta_i = 0$ , and the second transformation  $T_i^{i-1}$  between the links of two adjacent non-parallel joints is defined in Equation (2):

$$\begin{aligned}
 T_i^{i-1}(\alpha_{i-1}, a_{i-1}, d_i, \theta_i) &= \text{Rot}(x, \alpha_{i-1}) \text{Trans}(x, a_{i-1}) \text{Rot}(z, \theta_i) \text{Trans}(z, d_i) \\
 &= \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -d_i s\alpha_{i-1} \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & d_i c\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned} \tag{2}$$

Substituting the parameters of each joint into the homogeneous transformation matrix, the transformation matrix of the robot manipulator end actuator relative to the base is obtained, as shown in Equation (3):

$$T_{10}^0 = \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R & P \\ 0 & 1 \end{bmatrix} \tag{3}$$

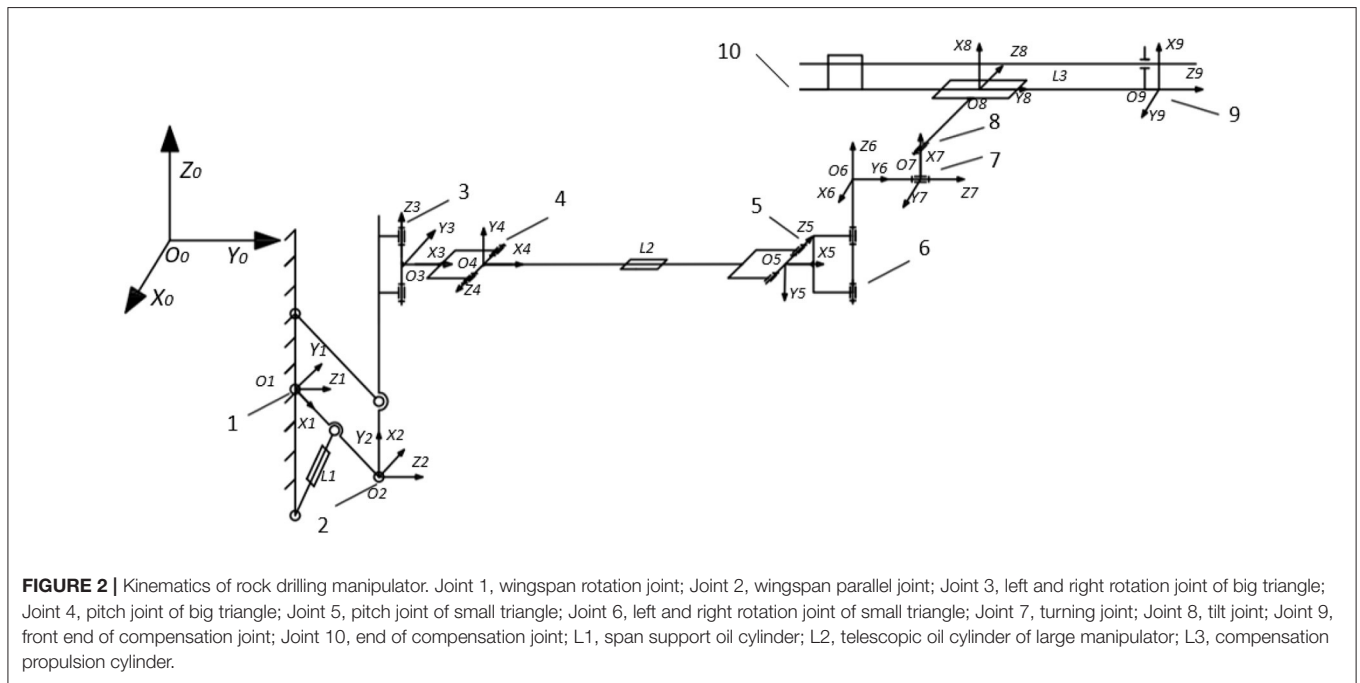
### 3. METHODOLOGY

#### 3.1. Parameter Identification of Geometric Error Compensation Model

The geometric error mainly refers to the error between the ideal kinematic model parameters and the parameters of the actual MDH modeling (Khan and Chen, 2011; Cui et al., 2012). Therefore, the geometric errors are composed of measurement errors of length and angles during modeling, the calibration errors of the sensor, the manufacturing errors, and assembly errors. For positive kinematics parameters,  $a_i$  represents the distance between the joint axes of two adjacent joints, and the main error caused by  $a_i$  is the actual machining error. The symbol  $\alpha_i$  represents the relative rotation angle of the two connected joints, and its main error comes from the coaxiality error in the assembly.  $d_i$  represents the relative distance between the two

joints on the common joint axis, which is mainly derived from the measurement error during modeling.  $\theta_i$  represents the rotation angle of two adjacent joints around the common axis; the error angle of two adjacent joints around the common axis; the error mainly comes from the observation error of the sensors.

The key role of the MDH method is to determine the axis of each joint. Therefore, by tracking and recording the trajectory of each joint's movement, the actual joint axis position of the robot can be established in the MDH model. The kinematic parameters of the manipulator are defined by error computation or estimation applying the posture measurement and position data of the end effector. In this article, the kinematic parameters are estimated by applying a differential evolution algorithm. The identified kinematics error parameters  $\Delta\alpha, \Delta a, \Delta d, \Delta\theta, \Delta\beta$  are defined as: Kinematic parameters are estimated applying differential evolution algorithm. The identified kinematics error parameters  $\Delta\alpha, \Delta a, \Delta d, \Delta\theta, \Delta\beta$  are defined as:



**TABLE 1 |** DH parameter of the rock drilling robot.

$i$	Variable	Variable range	$\alpha_{i-1}/(^{\circ})$	$a_{i-1}/\text{mm}$	$d_i/\text{mm}$	$\theta_i/(^{\circ})$	$\beta_i/(^{\circ})$
1	$\theta_1$	-15 to 60°	90°	200	250	$\theta_1$	0
2	$\theta_2$	15 to 60°	0	1,382	0	$-\theta_1$	0
3	$\theta_3$	30 to 150°	-90°	0	2,500	$\theta_3$	0
4	$\theta_4$	-15 to 60°	-90°	0	$\theta_4$	0	0
5	$L_2, \theta_5$	0~2,168 mm, -30 to 30°	0	5,000+ $L_2$	0	$\theta_5$	0
6	$\theta_6$	-75 to -125°	90°	145	0	$\theta_6$	0
7	$\theta_7$	-25 to -175°	-90°	0	900	$\theta_7$	0
8	$\theta_8$	15 to -20°	90°	240	580	$\theta_8$	0
9	$L_3$	0 to 1,778 mm	-90°	0	2,230+ $L_3$	0	0

$$F(\Delta\alpha, \Delta a, \Delta d, \Delta\theta, \Delta\beta) = \sum (e_d^2 + e_\delta^2) \quad (4)$$

where  $e_d$  denotes the error of the end position, which indicates the distance deviation between the predicted position of the model and the actual position. The symbol  $e_\delta$  is the error of the end attitude, which indicates the predicted rotation angle deviation between the model and the actual attitude on the x, y, and z axes. The observation prisms “r” and “s” are set at the front and end of the manipulator’s compensation cylinder. The displacement from the deflection deformation and other nonlinear factors are ignored in the geometric compensation. Then the attitude error at the end of the robotic manipulator  $[e_d^r \ e_\delta^r]$  can be obtained from the position error of the two observation  $[e_d^s \ e_\delta^s]$ . The geometric error model can be expressed as:

$$F(\Delta\alpha, \Delta a, \Delta d, \Delta\theta, \Delta\beta) = \sum (e_d^{r2} + e_\delta^{s2}) \quad (5)$$

### 3.2. Parallel Differential Evolution

The differential evolution algorithm is a heuristic algorithm that can be applied to solve the optimization problem through competition and cross-mutation. In order to identify the parameters of the rock drilling robot, the common differential evolution algorithm is proposed as the following five steps:

#### Step 1 Initialization

The initial population is generated by randomly sampling the feasible search space defined by joint bounds, as shown in Equation (6):

$$x_{ij}(0) = L_j + \text{rand}_{ij}(0, 1) (U_j - L_j) \quad (6)$$

Let  $n$  be the individual dimension, expressed as the value of the first individual of the generation population in the dimension, and the individual randomly generates an initial value that meets the constraints in each dimension:

$$x_{ij}(0) = L_j + \text{rand}_{ij}(0, 1) (U_j - L_j) \quad (7)$$

In Equation (6),  $U_j$  and  $L_j$  represent the upper and lower bounds of the individual on the dimension and  $\text{rand}_{ij}(0, 1)$  represents a random number between  $[0, 1]$  that obeys a uniform distribution.

#### Step 2 Mutation

There are multiple solutions for the parameter identification of the rock drilling robot error model, the robust DE/rand/1/bin evolution model is used to preserve the diversity of the search population (Mlakar et al., 2015). The specific operation is to select an individual with the highest fitness from the current population, randomly select three different individuals, and scale the difference between the two vectors and add it to another individual vector to obtain a new variable Individual (Pavelski et al., 2016):

$$v_i(t) = x_{p1}(t) + F(x_{p2} - x_{p3}) \quad (8)$$

where  $F$  is the scaling factor of DE, the value range is  $[0, 1]$ .

#### Step 3 Boundary constraint

The individual vector after the mutation operation may exceed the boundary of the search solution space, and the infeasible solution is converted to generate a new vector (Tian et al., 2020):

$$v_{ij}(t) = \begin{cases} \min \{U_j, 2L_j - v_{ij}(t)\}, & \text{if } v_{ij} < L_j \\ \max \{L_j, 2U_j - v_{ij}(t)\}, & \text{if } v_{ij} > U_j \end{cases} \quad (9)$$

#### Step 4 Crossover

The differential evolution algorithm crosses the reference vector and mutation vector to increase the diversity of the next generation population. The specific operations are as follows:

$$h_{ij}(t+1) = \begin{cases} v_{ij}(t+1), & \text{if } \text{rand}_j[0, 1] < Cr \text{ or } j = j_{\text{rand}} \\ x_{ij}(t), & \text{Otherwise} \end{cases} \quad (10)$$

Where:  $\text{rand}[0,1]$  represents the uniformly distributed random number on the  $j$ th dimension,  $C_r$  represents the crossover probability in the range of  $[0,1]$ ,  $j_{\text{rand}}$  represents a random number in  $1, 2, \dots, n$ .

#### Step 5 Selection

Based on the greedy selection mechanism, it compares the fitness of the mutated and cross-generated individual with the original individual and retains the highly adaptive individual as the next-generation population individual. The specific expression is as follows:

$$X_i(t+1) = \begin{cases} H_i(t+1), & \text{if } f(H_i(t+1)) > f(X_i(t)) \\ X_i(t), & \text{if } f(H_i(t+1)) < f(X_i(t)) \end{cases} \quad (11)$$

where  $f$  is the fitness function.

Since the rock drilling robot has three moving joints and six rotating joints, the dimension numbers of the error compensation model is 38. The high-dimensional solution requires a high number of populations and evolutionary algebra, which poses challenges to computer performance, algorithm model convergence, and computing speed. In order to accelerate the model convergence and increase the calculation speed under the same computing power input, a parallel structure of the differential evolution algorithm is proposed. The computing performance of the CPU can be enhanced by applying a

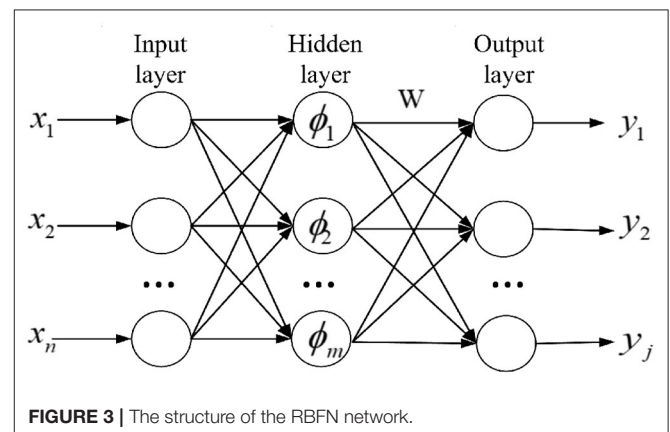


FIGURE 3 | The structure of the RBFN network.

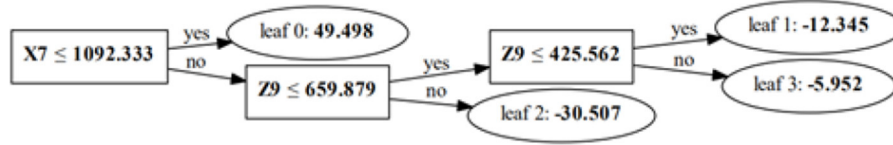


FIGURE 4 | One of the regression tree of lightGBM.

parallel differential evolution algorithm to calculate independent individual populations. Specifically, due to the independence of the sub-populations, multiple processes can be allocated through different computing cores of the CPU to calculate the fitness of multiple sub-populations parallel. The structure of the parallel differential evolution algorithm is shown in Figures 2–6. Set the kinematic parameter error compensation vector  $\Delta\alpha, \Delta a, \Delta d, \Delta\theta, \Delta\beta$  to be the population individual in the differential evolution algorithm. The sum of squared errors between the trajectory of the end of the robotic manipulator and the observation trajectory in the population is  $F$ .

### 3.3. Non-geometric Error Analysis

Due to the heavy load, large span, redundant joints, and hydraulic character of the rock drilling robot, the error shows a high degree of non-linearity. After geometric error compensation is performed on the positive kinematics model, there are still errors due to the non-linearity. The non-geometric errors are composed of regular errors and irregular errors. Regular errors refer to predictable errors, including elastic deformation due to changes in load, frictional force direction, structural changes, and the cumulative error caused by the transducer. The irregular errors mainly come from observation errors and system errors. The observation errors are related to the dynamic position capture accuracy of the total station, the observation elevation angle, and the measurement accuracy of the optical sensor; the system errors mainly come from the command error of the encoder and the signal transmission communication delay, vibration errors caused by kinetic energy changes, etc. Regular non-geometric errors are related to the time and space of the robot arm movement. Take the compensation joint L3 as an example. Moreover, changes in the relative speed and direction of joint motion affect the magnitude and direction of frictional resistance, and thereby the position error is changed.

Features are important conditions that affects the machine learning models. Important features are extracted from the original data for the use of algorithms and models (Yu et al., 2017; Wen et al., 2021). Different algorithms have different feature selections during supervised learning training due to their training methods and structural characteristics. To perform supervised learning training on non-geometric error compensation of rock drilling robots, various features are extracted according to different input feature vectors. The feature types are divided into three types:

(1) **Feature of the joint:** The value of each joint sensor is applied as a feature vector.

- (2) **Feature of positive kinematics model:** The positive kinematics of the robotic manipulator can project the posture of each joint from the sensor to the Cartesian three-dimensional coordinate, including the motion state, spatial changes, and transformations of the non-geometric errors.
- (3) **Feature of control system:** The feature of the control system refers to the signal to control the electro-hydraulic proportional valve, and the direction of joint, movement speed, and stroke distance. The hydraulic control is affected by the temperature, pressure, external load, and external environment of the hydraulic system.

### 3.4. RBFN

RBFNN is a three-layer neural network that can quickly approximate the nonlinear function to overcome local minimums. There are three layers in RBFN, namely including an input layer, a hidden layer, and an output layer, as shown in Figure 3.

The relationship of the RBFN from vector input to output is described as:

$$y_i = f_i(x) = \sum_{k=1}^m \omega_{ik} \phi_k(x, c_k) = \sum_{k=1}^m \omega_{ik} \phi_k(\theta \|x - c_k\|) \quad (12)$$

where  $x$  is the input vector,  $\omega$  is the output layer connection weight,  $\phi$  is the hidden layer basis function, and  $c_k$  is the basis function center of the  $k$ th neuron node

The radial basis kernel function is defined as Gaussian radial basis function:

$$\phi_k(\|\theta(x - c_k)\|) = \exp\left[-\frac{(x - c_k)^T \cdot \theta \cdot (x - c_k)}{2\sigma_k^2}\right] \quad (13)$$

where  $\sigma_k$  is the expansion width of the radial basis function of the first neuron in the hidden layer.

Features are composed of the joint sensor value and the three-dimensional coordinates calculated through positive kinematics. In this article, the input of the RBFN model is normalized “joint sensor value (9-dimensional)” and “positive kinematics model parameters (27-dimensional)” which is 36-dimensional in total, and the output is the predicted three-dimensional coordinates. When  $k = 100$  and  $\sigma = 0.05$ , it is validated that the model shows the best result through experiments.

### 3.5. LightGBM

LightGBM is a fast, distributed, and high-performing framework based on gradient boosting decision tree (GBDT), which has

wide applications in for decades for due to its superior predictive performance. Combining gradient boosting and decision tree, lightGBM has a strong expressive ability, good training effect, and can effectively prevent overfitting by controlling the growth of the tree. It can solve continuous and discrete values flexibly and is robust to outliers. The process is mainly composed of three steps, outlined as below:

**Step 1** The training data set is given as:  $P = \{(x_1, y_1), \dots, (x_2, y_2)\}$ , let the loss function be  $L(y, \gamma)$ , the learner is initialized:

$$f_0(x) = \arg \min_{\gamma} \sum_{i=1}^N L(y_i, \gamma) \quad (14)$$

**Step 2**  $M$  base learners are iteratively generated

The negative gradient of each samples  $i = 1, 2, \dots, N$  is calculated as:

$$r_{mi} = - \left[ \frac{\partial L(y_i, f(x_i))}{\partial f(x_i)} \right]_{f(x)=f_{m-1}(x)} \quad (15)$$

$(x_i, r_{mi}) (i = 1, 2, \dots, N)$  is utilized as the training data for the next regression tree. The leaf node area  $R_{(jm)}$  of the  $m$ -th tree  $f_m(x)$ ,  $j = 1, 2, \dots, J$ ,  $J$  is the number of leaf nodes of the regression tree. The best fit is calculated as

$$\gamma_{jm} = \arg \min_{x_i \in R_{jm}} L(y_i, f_{m-1}(x_i) + \gamma) \quad (16)$$

**Step 3** The strong learner is generated as:

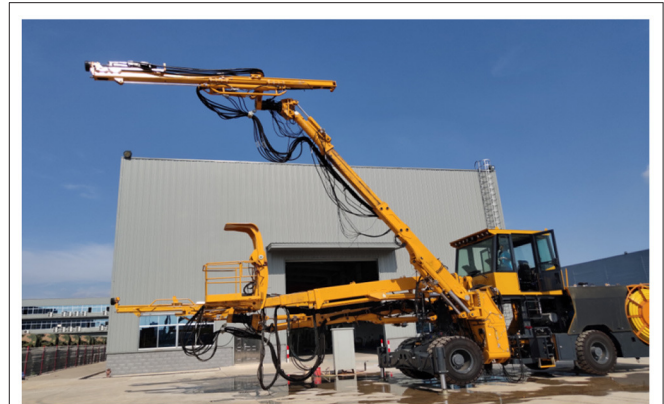
$$f(x) = f_M(x) = f_0(x) + \sum_{m=1}^M \sum_{j=1}^J \gamma_{jm} I(x \in R_{jm}) \quad (17)$$

The endpoint error compensation is used as the training label for supervised learning; its training features have a total of 66 dimensions, which are “feature of the joint (9 dimensions)” + “feature of positive kinematics model (27 dimensions)” + “feature of control system (30 dimensions).”  $K$ -fold cross-validation method is applied. The test data  $D$  is divided into  $k$  parts in equal proportions. One of the parts is applied as the verification set. The other  $K - 1$  parts of data are used as the training data. After performing  $K$  times of training, the average result of the  $K$  training experiments is used as the final result. When the model stability is low, increasing the value of  $K$  can achieve better results.

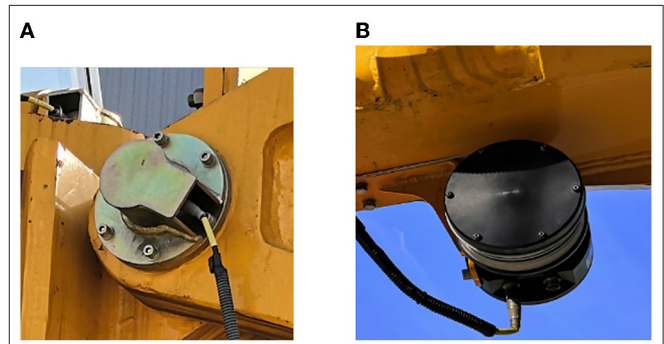
## 4. EXPERIMENTAL VALIDATION AND RESULTS

### 4.1. Experimental Set-Up

In order to achieve accurate position compensation for the rock drilling robot, the error compensation experiment is performed. In this experiment, the SWDT82 rock drilling robot developed by Sunward Intelligent Co., Ltd, as shown in **Figure 5**, has a maximum manipulator length of 12.8 m and a wingspan



**FIGURE 5** | SWDT82 rock drilling robot.



**FIGURE 6** | The sensors applied in the rock drilling robot, **(A)** a joint angle sensor, **(B)** is a length transducer.



**FIGURE 7** | Experiment of the rock drilling robot.

of 5.2 m. The maximum coverage section width of the dual-manipulator rock drilling robot is 17.65 m, the maximum coverage section height is 13.38 m, and the maximum invert depth is 3.31 m.

The rock drilling robot is equipped with a three length transducer and seven angle sensors as shown in **Figure 6**. The angle sensors are the rotary encoder of POSITAL's model IXARC, and its theoretical measurement error is  $0.1^\circ$ ; the length transducer is TRANSTRONIC's model 022441, and the theoretical error of measurement linearity of this sensor is  $0.1\%$  mm. The observation instrument in the experiment is a Topcon laser total station, which has functions such as motor tracking, automatic sighting, and Bluetooth transmission.

### 4.2. Experimental Results

To compensate for the geometric error of the rock drilling robot, the motion of a single joint is carried out. The position and posture of the rock drilling robot are collected through Topcon total station transmitting to the master computer using Bluetooth as shown in **Figure 7**. The data of the angle sensors and length transducers are transmitted to the master computer  $w, b, r,$  and  $s$  corresponding to the joint 3, the joint 5, the front of the compensation cylinder 9, and the end position of the compensation cylinder 10, as shown in **Figure 8**. Moreover, the end posture of the rock drilling manipulator is determined by the position of observation point  $r$  and  $s$ .

So as to identify the geometric error of the manipulator, single-joint motion of the manipulator is carried out. After substituting each observation point, obtained from sensors, into the forward kinematics model for calculation, the absolute errors can be obtained. The absolute average error of each observation point is  $e_{r0}^{\bar{}}: e_{s0}^{\bar{}}: e_{b0}^{\bar{}}: e_{w0}^{\bar{}} = 323.59: 477.53: 895.52: 930.88$  (mm).

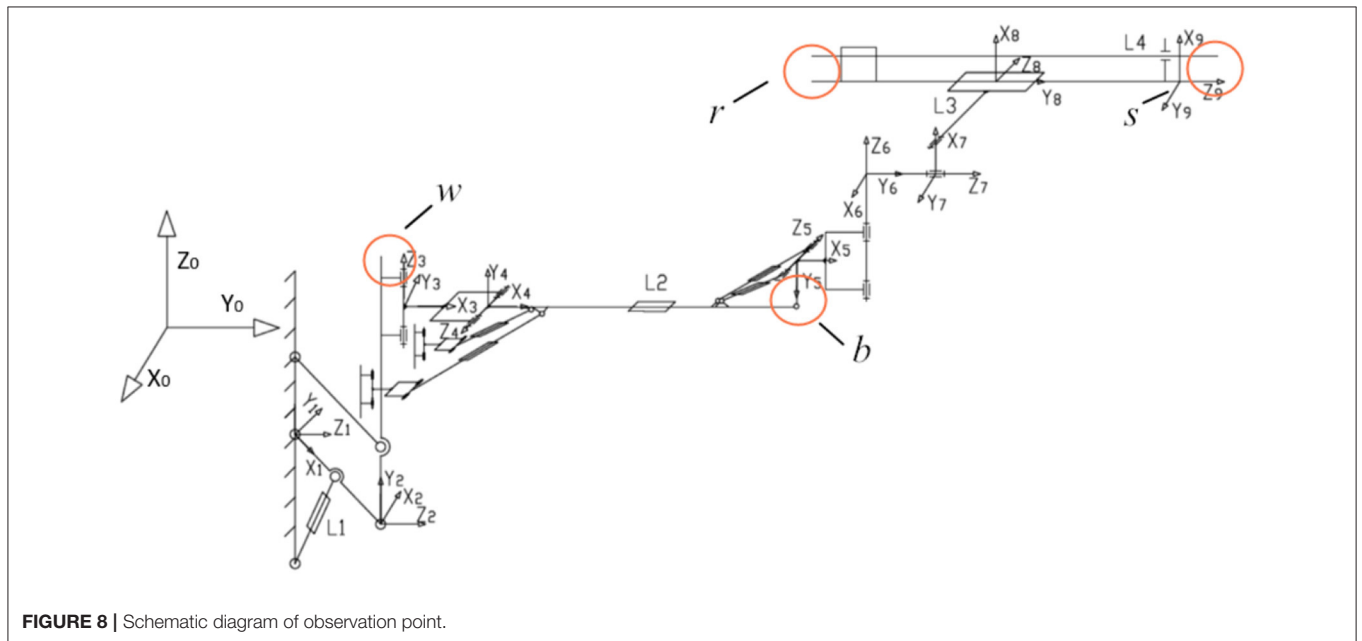
Afterwards, the multi-joint motion experiment is carried out. Trajectory data of the observation points of rock drilling robot motion  $r, s, b, w$  is applied for error analysis. In the multi-joint movement, the absolute average error of each observation

point data set  $e_{r0}^{\bar{}}: e_{s0}^{\bar{}}: e_{b0}^{\bar{}}: e_{w0}^{\bar{}} = 355.61: 783.17: 1085.32: 1288.916$  (mm).

In order to identify the forward kinematics of the manipulator, the differential evolution method is applied to compensate for the geometric error. This experiment not only explores the error compensation effect of the error model method but also explores the influence of the amount of data on the convergence of the parameter identification of the error model method. Due to the high-dimension of the model, the population of the differential evolution algorithm is set to 30,000. To avoid premature convergence of parameter optimization, the differential cross recombination probability  $C_r$  is set to 1, the scaling factor  $F = 0.5$ , and the number of evolutions is set to 400. To improve the performance of the model, the data set is shuffled and substituted into the model multiple times to obtain the optimal solution. To optimize the error model, the joint constraints are increased. Set the end trajectory of each observation point as  $L'_w, L'_b, L'_s, L'_r$ , the trajectory observation point set is  $P'$  and the model prediction point set are  $(X', Y', Z')$ . Then the position error of each observation position are  $e_w^d, e_b^d, e_s^d, e_r^d$ . The average absolute error  $F$  of the optimal error model with ideal multi-joint constraints at each observation point is expressed as:

$$\begin{cases} F_w = \min(e_w^{d2}) \\ F_b = \min(e_b^{d2}) \\ F_s = \min(e_s^{d2}) \\ F_r = \min(e_r^{d2}) \end{cases} \quad (18)$$

The solutions of  $F_w, F_b, F_s,$  and  $F_r$  are difficult to satisfy simultaneously, and the reduction of the error of some joint clusters requires the degradation of the error fitting effect of the other joint clusters, resulting in the Pareto solution. And new observation errors are introduced by adding trajectory



**FIGURE 8** | Schematic diagram of observation point.



constraints, which reduce the overall accuracy of the model. In order to solve the above problems, an equal amount of data in the model calculation are applied for each joint. Moreover, a weight coefficient is assigned to each joint, and the functions are obtained by linearly combining the multi-objective functions:

$$F(\Delta\alpha, \Delta a, \Delta d, \Delta\theta, \Delta\beta) = \frac{\mu_r e_r^{d^2}}{n_r} + \frac{\mu_s e_s^{d^2}}{n_s} + \frac{\mu_b e_b^{d^2}}{n_b} + \frac{\mu_w e_w^{d^2}}{n_w} \quad (19)$$

where  $\mu_r, \mu_s, \mu_b,$  and  $\mu_w$  are the weight coefficients of the multi-objective error constraint, and  $n_r, n_s, n_b,$  and  $n_w$  are the numbers

**TABLE 2** | Parameters of positive kinematics compensation.

<i>i</i>	$\Delta\alpha_{i-1}/(^{\circ})$	$\Delta a_{i-1}/\text{mm}$	$\Delta d_{i-1}/\text{mm}$	$\Delta\theta_{i-1}/(^{\circ})$	$\Delta\beta_{i-1}/(^{\circ})$
1	3.5932	266.49	-37.92	3.9988	0
2	0.3292	-91.55	85.62	2.3400	5.3136
3	5.9856	90.88	96.04	2.9599	0
4	-5.7576	95.11	369.38	5.1696	-2.9444
5	-3.1405	123.58	230.58	1.1894	0
6	-0.5786	-253.95	-33.82	-0.5875	0
7	1.0460	269.17	225.12	4.6678	0
8	-1.0995	376.07	24.36	2.8406	0
9	3.5932	266.49	-37.92	3.9988	0

**TABLE 3** | The mean absolute error of each experiment.

	$\bar{e}_r$ (mm)	$\bar{e}_s$ (mm)	$\bar{e}_b$ (mm)	$\bar{e}_w$ (mm)
Single joint	382.58	396.04	466.92	263.21
Multi joint	389.08	334.58	290.45	44.83
Optimal method	306.82	320.12	298.69	70.37

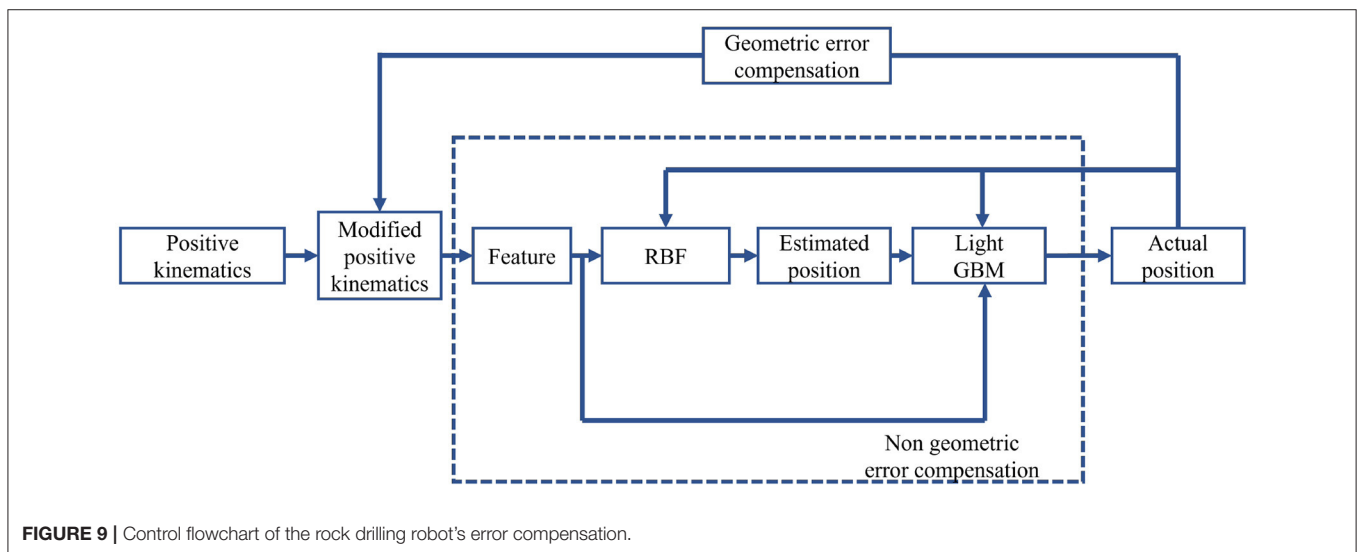
of each data set. The parameters of forward kinematics of the optimal method are obtained shown in as **Table 2**.

The average absolute error can be calculated from the compensated positive kinematics. A parallel differential evolution algorithm is applied to move a single joint as the first experiment. Then, a parallel differential evolution algorithm is applied for moving multi-joints applying as the second experiment. Ultimately, the optimal differential evolution algorithm is applied as the third experiment. Consequently, the distribution of errors is shown in **Table 3**.

Applying the optimal method of parallel differential evolution, the errors of the end effectors are compensated, and the best performance is achieved compared to the parallel differential evolution algorithm.

The overall control flowchart is shown in **Figure 9**. Firstly, the modified positive kinematics is obtained through an optimal parallel differential evolution algorithm to compensate for the geometric error. Afterward, the features are applied to train RBFN. The estimated position and features are combined to train light GBM. The non-geometric errors are compensated through RBFN and light GBM. The top 10 features are shown in **Figure 10**.  $X_i, Y_i, Z_i$  ( $i=1, 2..$ ) are the Cartesian coordinates of the joint  $i$ ,  $orient3$  is the control current of the hydraulic valve,  $Propeller\_Flip\_Angle$  is the sensor of the flip joint,  $Z_{dabi}, Y_{dabi}$  denote the Z coordinate and Y coordinate of the big manipulator separately, Y coordinate. The hydraulic valve current control ranks 7th in feature importance as a system feature, indicating that the system feature has an important contribution for the model.

The absolute error of the endpoint is obtained for multiple experiments in **Figure 11**. All of the experiments are carried out by applying the same positive kinematics from the optimal parallel differential evolution. The average error of the RBFN is 41.5 mm, while the standard error of the RBFN is 33.5 mm.



**FIGURE 9** | Control flowchart of the rock drilling robot's error compensation.

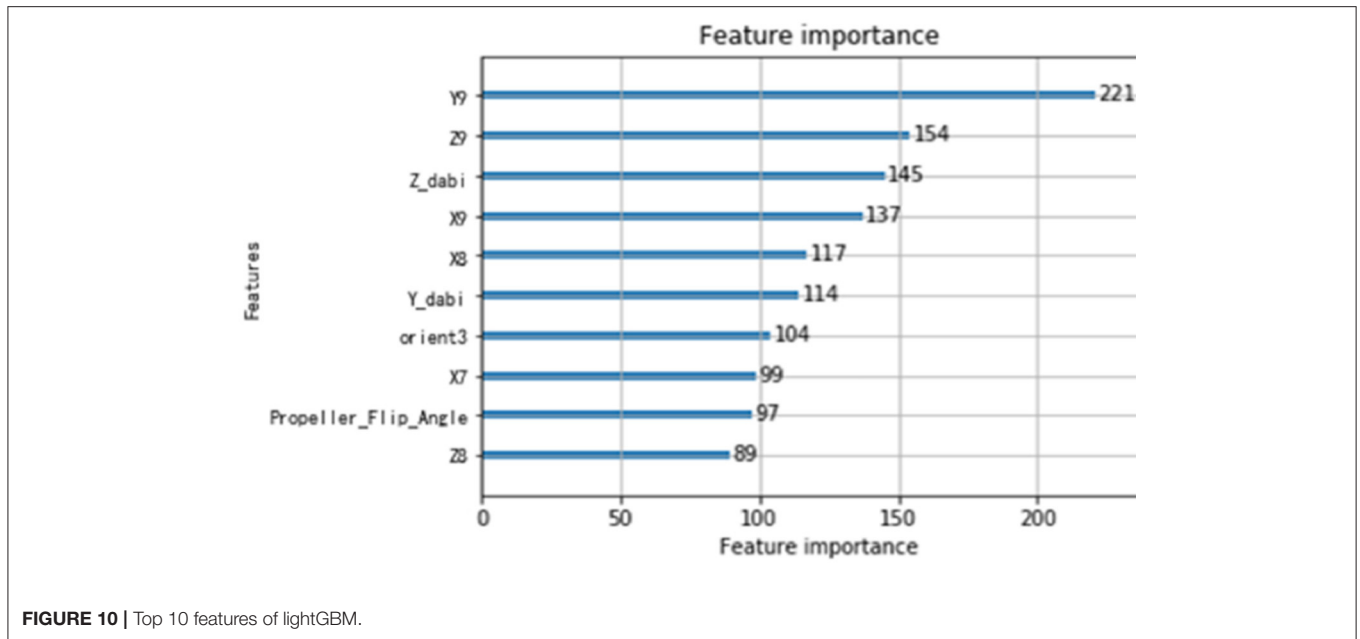


FIGURE 10 | Top 10 features of lightGBM.

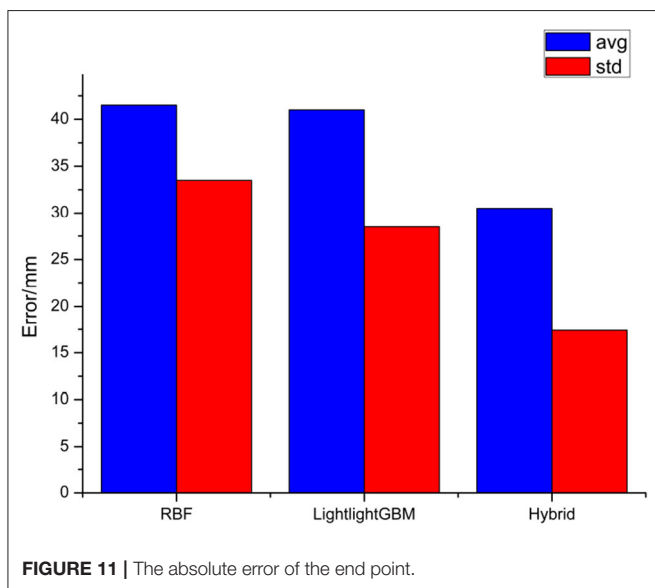


FIGURE 11 | The absolute error of the end point.

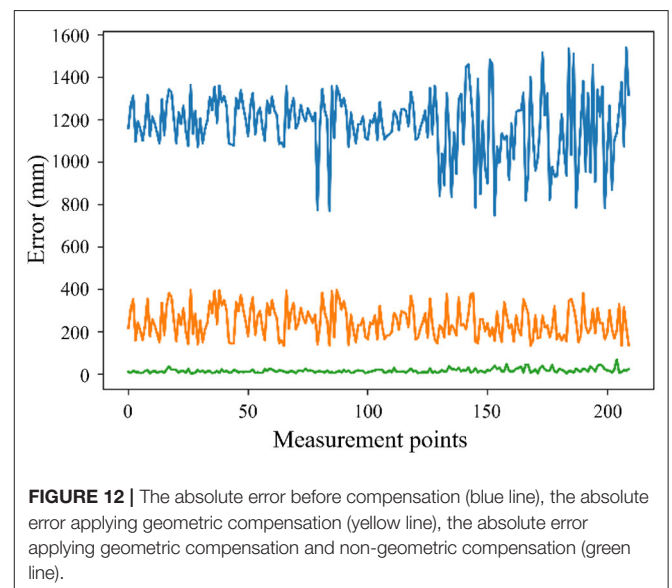


FIGURE 12 | The absolute error before compensation (blue line), the absolute error applying geometric compensation (yellow line), the absolute error applying geometric compensation and non-geometric compensation (green line).

Moreover, the average error of the lightGBM is 41.0 mm, while the standard error of the RBFN is 28.5 mm. Moreover, the average error of the hybrid method is 30.5 mm, while the standard error of the hybrid method is 17.4 mm. It can be concluded that the hybrid method can achieve a good performance.

A validation experiment is carried out to verify the effectiveness of the proposed method as shown in Figure 12. Two hundred points collected along the trajectory are analyzed in two error compensation methods. The blue line is the absolute error before compensation. The yellow line is the absolute error applying the parallel difference algorithm for geometric compensation. The green line is the absolute error applying RBFN combined with light GBM for geometric compensation

and non-geometric compensation. It can be concluded that the error of the rock drilling robot is compensated effectively by applying the proposed method, which meets the practical application needs of rock drilling robots.

## 5. DISCUSSION AND SUMMARY

Geometric error compensation and non-geometric error compensation methods are combined in this article. Firstly, the positive kinematics model of the rock drilling robot is established by the MDH method. Then, a parallel differential evolution algorithm is designed to identify the forward kinematic parameters to compensate for the geometric error. Single joint

and multi-joint experiments are carried out to analyze the error. By analyzing non-geometric error through experimental trajectories, non-geometric error compensation based on RBF neural network and lightGBM is proposed. Moreover, feature engineering for the rock drilling robot is analyzed based on the robotic kinematics model to achieve the optimization of the non-geometric error compensation model. Finally, the experiment is carried out to validate the performance of the proposed method, which meets the precision requirements of rock drilling robots. This article has shown that the addition of positive kinematic parameters and control signal parameters as feature engineering can greatly improve the learning ability of machine learning. Therefore, features such as speed, oil pressure, and artificial deflection calculation can be considered in the research of machine learning to further improve the learning ability and generalization of the model. And the deflection deformation of the manipulators can be modeled and analyzed in the field of mechanical materials to enhance positional accuracy.

## DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

## REFERENCES

- Breiman (2001). Random forests. *Mach. Learn.* 45, 5–32. doi: 10.1023/A:1010933404324
- Chen, C.-H., Tanaka, K., and Funatsu, K. (2018). Random forest model with combined features: a practical approach to predict liquid-crystalline property. *Mol. Informatics* 38:1800095. doi: 10.1002/minf.201800095
- Chen, D., Wang, T., Yuan, P., Sun, N., and Tang, H. (2019). A positional error compensation method for industrial robots combining error similarity and radial basis function neural network. *Meas. Sci. Technol.* 30:125010. doi: 10.1088/1361-6501/ab3311
- Chen, D., Yuan, P., Wang, T., Ying, C., and Tang, H. (2018). A compensation method based on error similarity and error correlation to enhance the position accuracy of an aviation drilling robot. *Meas. Sci. Technol.* 29:085011. doi: 10.1088/1361-6501/aacd6e
- Chen, P., Niu, A., Jiang, W., Liu, D., Ma, B., and Bao, T. (2019). Air pollutant prediction: comparisons between LSTM, light GBM and random forests. *J. Environ. Protect. Ecol.*
- Cui, G., Yong, L., Li, J., Gao, D., and Yao, Y. (2012). Geometric error compensation software system for CNC machine tools based on NC program reconstructing. *Int. J. Adv. Manufact. Technol.* 63, 169–180. doi: 10.1007/s00170-011-3895-0
- Huang, G., Zhu, Q., and Siew, C. (2004). “Extreme learning machine: a new learning scheme of feedforward neural networks,” in *IEEE International Joint Conference on Neural Networks* (Budapest), 985–990.
- Jiang, Z., Huang, M., Tang, X., and Guo, Y. (2021). A new calibration method for joint-dependent geometric errors of industrial robot based on multiple identification spaces. *Robot. Comput. Integr. Manufact.* 71:102175. doi: 10.1016/j.rcim.2021.102175
- Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., et al. (2017). “LightGBM: a highly efficient gradient boosting decision tree,” in *Advances in Neural Information Processing Systems* (Beijing).
- Khan, A. W., and Chen, W. (2011). A methodology for systematic geometric error compensation in five-axis machine tools. *Int. J. Adv. Manufact. Technol.* 53, 615–628. doi: 10.1007/s00170-010-2848-3

## AUTHOR CONTRIBUTIONS

XZ contributed to the development of the simulation engine of the platform under the guidance of JH, YZ, and GB. JD and PL contributed to experimental acquisition and algorithm verification. XZ and WB contributed to analysis of the data from simulation and physical experiments. All authors contributed to writing, reviewing, and proofreading the manuscript.

## FUNDING

This research was supported by the National Natural Science Foundation of China (52105127) and the Natural Science Foundation of Zhejiang Province, PRC (LGG22E050025).

## ACKNOWLEDGMENTS

The authors would like to thank Dr. Hongqiang Zhao and Dr. Xihua Xie for their advice and guidance while designing the rock drilling robot, Shurong Gao and Dr. Daqing Zhang for sharing their expertise in manufacturing and controls. We would also like to thank Dr. Changsheng Liu, Peng Hu, and all other members of the Sunward Intelligent Equipment Company for their assistance with the experiments.

- Mlakar, M., Petelin, D., Tušar, T., and Filipič, B. (2015). GP-demo: Differential evolution for multiobjective optimization based on Gaussian process models. *Eur. J. Oper. Res.* 243, 347–361. doi: 10.1016/j.ejor.2014.04.011
- Molfino, R. M., Razzoli, R. P., and Zoppi, M. (2008). Autonomous drilling robot for landslide monitoring and consolidation. *Autom. Constr.* 17, 111–12122. doi: 10.1016/j.autcon.2006.12.004
- Naghibi, S. A., Hashemi, H., Berndtsson, R., and Lee, S. (2020). Application of extreme gradient boosting and parallel random forest algorithms for assessing groundwater spring potential using DEM-derived factors. *J. Hydrol.* 589:125197. doi: 10.1016/j.jhydrol.2020.125197
- Ocak, I., and Bilgin, N. (2010). Comparative studies on the performance of a roadheader, impact hammer and drilling and blasting method in the excavation of metro station tunnels in Istanbul. *Tunnel. Undergr. Space Technol.* 25, 181–187. doi: 10.1016/j.tust.2009.11.002
- Park, J., and Sandberg, I. W. (1991). Universal approximation using radial-basis-function networks. *Neural Comput.* 3, 246–257. doi: 10.1162/neco.1991.3.2.246
- Pavelski, L. M., Delgado, M. R., Almeida, C. P., Gonçalves, R. A., and Venske, S. M. (2016). Extreme learning surrogate models in multi-objective optimization based on decomposition. *Neurocomputing* 180, 55–67. doi: 10.1016/j.neucom.2015.09.111
- Praveen, K. M., and Denis, A. S. (2018). Artificial neural network based geometric error correction model for enhancing positioning accuracy of a robotic sewing manipulator. *Proc. Comput. Sci.* 133, 1048–1055. doi: 10.1016/j.procs.2018.07.069
- Qi, W., Wang, N., Su, H., and Aliverti, A. (2021). DCNN based human activity recognition framework with depth vision guiding. *Neurocomputing* 486, 261–271. doi: 10.1016/j.neucom.2021.11.044
- Sewdien, V., Preece, R., Torres, J. R., Rakhshani, E., and van der Meijden, M. (2020). Assessment of critical parameters for artificial neural networks based short-term wind generation forecasting. *Renew. Energy* 161, 878–892. doi: 10.1016/j.renene.2020.07.117
- Su, H., Mariani, A., Ovr, S. E., Menciassi, A., Ferrigno, G., and De Momi, E. (2021). Toward teaching by demonstration for robot-assisted minimally invasive surgery. *IEEE Trans. Autom. Sci. Eng.* 18, 484–494. doi: 10.1109/TASE.2020.3045655

- Su, H., Qi, W., Hu, Y., Karimi, H. R., Ferrigno, G., and Momi, E. D. (2022). An incremental learning framework for human-like redundancy optimization of anthropomorphic manipulators. *IEEE Trans. Indus. Inform.* 18, 1864–1872. doi: 10.1109/TII.2020.3036693
- Su, H., Sandoval, J., Veyres, P., Poisson, G., Ferrigno, G., and De Momi, E. (2018). Safety-enhanced collaborative framework for tele-operated minimally invasive surgery using a 7-DOF torque-controlled robot. *Int. J. Control Autom. Syst.* 16, 2915–2923. doi: 10.1007/s12555-017-0486-3
- Svetnik, V., Liaw, A., Tong, C., Culberson, J., Sheridan, R., and Feuston, B. (2003). Random forest: a classification and regression tool for compound classification and QSAR modeling. *J. Chem. Inform. Comput. Sci.* 43, 1947–1958. doi: 10.1021/ci034160g
- Tian, Y., Zheng, X., Zhang, X., and Jin, Y. (2020). Efficient large-scale multiobjective optimization based on a competitive swarm optimizer. *IEEE Trans. Cybern.* 50, 3696–3708. doi: 10.1109/TCYB.2019.2906383
- Wen, Z., Zhou, C., Pan, J., Nie, T., Jia, R., and Yang, F. (2021). Froth image feature engineering-based prediction method for concentrate ash content of coal flotation. *Miner. Eng.* 170:107023. doi: 10.1016/j.mineng.2021.107023
- Yao, M., Zhu, Y., Li, J., Wei, H., and He, P. (2019). Research on predicting line loss rate in low voltage distribution network based on gradient boosting decision tree. *Energies* 12:2522. doi: 10.3390/en12132522
- Yu, S., Chen, H., and Brown, R. A. (2017). Hidden Markov model-based fall detection with motion sensor orientation calibration: a case for real-life home monitoring. *IEEE J. Biomed. Health Inform.* 22, 1847–1853. doi: 10.1109/JBHI.2017.2782079
- Yuan, P., Chen, D., Wang, T., Cao, S., Cai, Y., and Xue, L. (2018). A compensation method based on extreme learning machine to enhance absolute position accuracy for aviation drilling robot. *Adv. Mech. Eng.* 10:1687814018763411. doi: 10.1177/1687814018763411
- Zeng, Y., Tian, W., Li, D., He, X., and Liao, W. (2017). An error-similarity-based robot positional accuracy improvement method for a robotic drilling and riveting system. *Int. J. Adv. Manuf. Technol.* 88, 2745–2755. doi: 10.1007/s00170-016-8975-8
- Zhang, F., Shang, W., Li, G., and Cong, S. (2021). Calibration of geometric parameters and error compensation of non-geometric parameters for cable-driven parallel robots. *Mechatronics* 77:102595. doi: 10.1016/j.mechatronics.2021.102595
- Zhou, K., Hu, Y., Pan, H., Kong, L., Liu, J., Huang, Z., et al. (2020). Fast prediction of reservoir permeability based on EFS-LightGBM using direct logging data. *Meas. Sci. Technol.* 31:045101. doi: 10.1088/1361-6501/ab4a45
- Zhu, Y., Zhuang, F., Wang, J., Ke, G., Chen, J., Bian, J., et al. (2020). “Deep subdomain adaptation network for image classification,” in *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 32 (IEEE), 1713–1722.

**Conflict of Interest:** WB was employed by Zhejiang Jinbang Sports Equipment Co. Ltd. PL and YZ were employed by Sunward Intelligent Equipment Company, Ltd.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

**Publisher’s Note:** All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Zhou, Bai, He, Dai, Liu, Zhao and Bao. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.