www.s2is.org
International Journal on Smart Sensing and Intelligent Systems
ISSN 1178-5608

ERROR MODELING AND COMPENSATION OF 3D SCANNING ROBOT SYSTEM BASED ON PSO-RBFNN

Jianhong Qi，Jinda Cai

Mechanical Automation Research Institute

University of Shanghai for Science and Technology, 200093

Shanghai, China

Emails: qi.jianhong@163.com

Submitted: Nov.30, 2013 ***Accepted: Apr.17, 2014*** ***Published: June 1, 2014.***

Abstract- In order to improve the measurement accuracy of three-dimensional (3D) scanning robot, a method of 3D scanning robot system error modeling and compensation based on particle swarm optimization radial basis function neural network (PSO-RBFNN) is proposed to achieve intelligent compensation of measurement error. The structure, calibration and error modeling process of 3D scanning robot system are mainly described. Cleverly using the iterative closest point (ICP) algorithm to construct input and output data pairs of neural network, and the specific process of error modeling using PSO-RBFNN is given. Finally, through the actual experiment we test and verify the correctness and effectiveness of the proposed error modeling and compensation method by measuring the distance between the centers of two standard balls. Experimental results show: the proposed error model and the compensation method can effectively compensate the measurement errors and improve the accuracy of the 3D scanning robot system.

Index terms: 3D scanning robot system, 6 degree of freedom (DOF) robot, line-structured light sensor, particle swarm optimization radial basis function neural network (PSO-RBFNN), error model, error compensation

I. INTRODUCTION

Modern industrial production needs more and more measurement of the product surface profile, three-dimensional size and various molds free form. While the shape of the product is complex and diverse in the meantime measuring working period decreases significantly, the traditional contact measurement methods cannot meet the demands of rapid measurement [1].Laser vision measurement as a non-contact measurement method, with many advantages of non-destructive, fast, online measurement, anti-interference ability, etc. It has been widely used in the surface shape inspection, reverse engineering [2].In recent years, with the development of optics and microelectronics technology, 3D scanning robot has been developed rapidly. It is convenient for access to three-dimensional information of objects by multi-angle and multi-faceted manner. It has better applications in the universe exploration and robot visual aspects.

This paper describes how to construct 3D scanning robot system by using the line-structured light sensor that has calibrated internal and external camera calibration parameters and the laser plane parameters and a self-developed NCD-1001-type six degrees of freedom(6-DOF) industrial robot[3]. On the basis of the above, robot system is calibrated. If the 3D scanning robot system would be used in digital manufacturing process quality inspection and quality control [4], improving the measurement accuracy of the system is the core problem. In order to improve the measurement accuracy further, we can use particle swarm optimization radial basis function neural network (PSO-RBFNN) to build error model and compensation scheme of 3D scanning robot system, by scanning a standard piece to obtain systematic measurement error. We also can compensate the measurement error by using the proposed measurement error network model in the measurement process to improve the system's accuracy. Through standard ball scan experiment, it shows that the method can compensate the measurement errors to some extent and improve system accuracy. Thus it verifies the correctness and effectiveness of the method.

II. THE WORKING PRINCIPLE OF THE 3D SCANNING ROBOT SYSTEM

3D scanning robot system based on line-structured light vision as shown in figure 1, the main components are as follows: ① 6-DOF robot and its controller; ② line-structured light sensor (fixed on the robot); ③ laser brightness controller; ④ Host computer and software systems; ⑤ workbench and the work piece to be scanned.

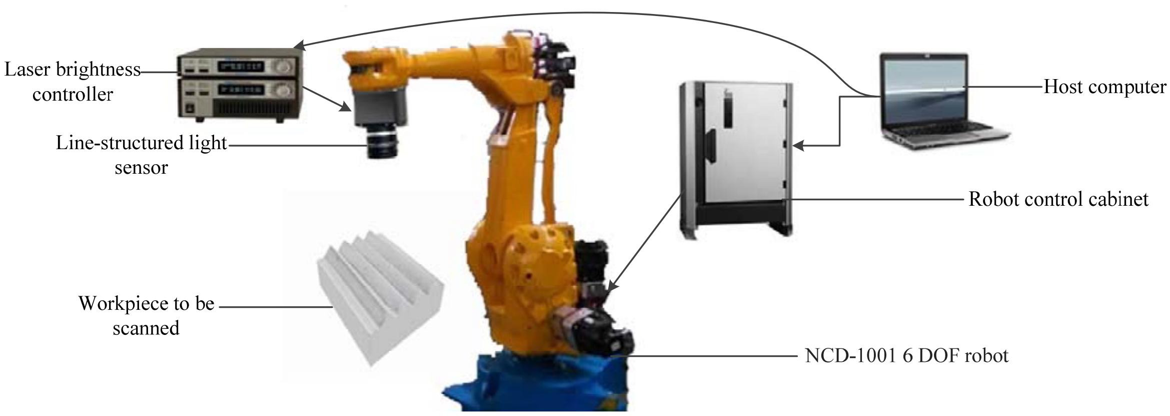


Figure1. Three-dimensional scanning robot system

Displacement system is mainly completed by the 6-DOF robot, if necessary, with a high-precision rotary table. NCD-1001 6-DOF industrial robot (figure 2) is the major component of the 3D scanning system, which is not only execution device, but also measuring device. Its accuracy plays a major role in the entire system accuracy [5]. Theoretically, the 6-DOF robot end can reach arbitrary position of its work space in any posture, so that it can obtain the three-dimensional coordinate information of complex free curved surface.

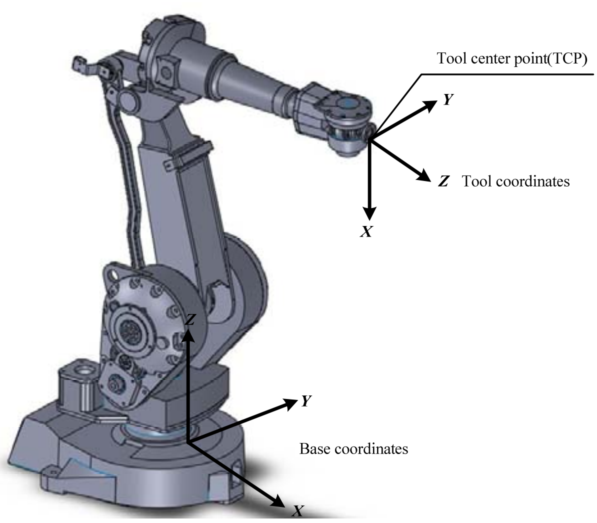


Figure 2. NCD-1001 six degrees of freedom (6-DOF) industrial robot

The working principle of 3D scanning robot system as follows: built-semiconductor laser of line-structured light sensor is utilized for emitting a laser to the work piece surface. The CCD camera and laser machine exist a certain angle to shoot the laser pieces of information within a certain range, through the image grabber, it transmits the captured information to the computer. The computer uses of corresponding image processing method to extract the laser stripe information. After extracting the light two-dimensional information, we can obtain the three-dimensional coordinates of the laser line in the sensor coordinate system with the line-structured light vision sensor mathematical model which obtained through calibration. Then it converts the three-dimensional coordinate into the flange coordinate system by the hand-eye matrix and calculates the depth and vertical information of the work piece [6]. Then the 6-DOF robot drives laser and camera will do the lateral movement, through computer information extraction to motor motion control card getting laterally information of the work piece. Finally we obtain the 3D point cloud space coordinates of the work piece surface through the robot kinematics model and calculation. Then the machine conveys point cloud data to the computer via the data post-processing software having done the smoothing filter and surface reconstruction and a series of operations for the obtained three-dimensional discrete point cloud. Thus completes of three-dimensional numerical modeling of the work piece is scanned. The working principle of the 3D scanning robot system is illustrated in figure 3.

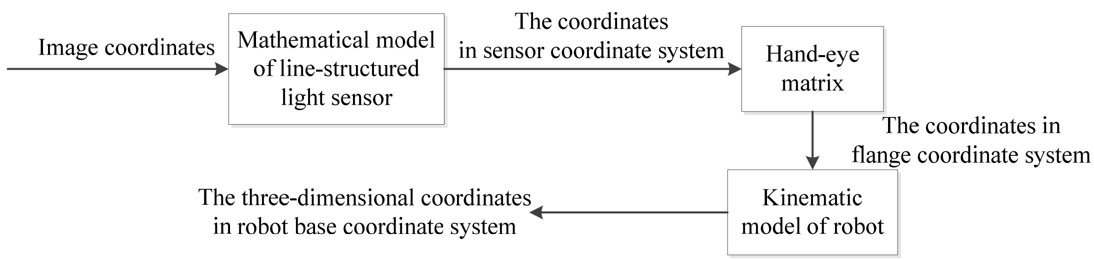


Figure3. Work principle of three-dimensional scanning robot system

III CALIBRATION OF THE 3D SCANNING ROBOT SYSTEM

Robot base coordinate system is the robot base coordinate system (Base).The end of the robot tool coordinate system is called $Tool_0$ (figure 2). Calibration of the 3D scanning robot is to determine the relationship between the sensor coordinate system and rotation matrix R_s and translation vector T_s of 3D scanning robot base coordinate systems [7].

1) To calibrate rotational relationship R_s by constraining the space point

For a point that has fixed position with $Tool_0$ coordinate system, the relationship between its coordinates (x_t, y_t, z_t) under the $Tool_0$ and its coordinates (x_L, y_L, z_L) under the sensor coordinate system need satisfy following equation:

$$\begin{bmatrix} \boldsymbol{X}_t \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{R_0} & \boldsymbol{T_0} \\ 0 & 1 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \boldsymbol{R_0} & \boldsymbol{T_0} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{X}_L \\ 1 \end{bmatrix} \tag{1}$$

In the equation (1): X_t is the coordinates (x_t, y_t, z_t)of the fixed point under the $Tool_0$ coordinate system, X_L is the recovered coordinate from sensor coordinate system of the fixed point (i.e., the coordinate (x_L, y_L, z_L) relative to scanner coordinate), R_0 is the rotation matrix in the $Tool_0$ (i.e., the rotation matrix of robot end coordinate system relative to robot base coordinate system), T_0 is the translation matrix in $Tool_0$, Rs and Ts are rotation and translation matrix of sensor coordinate system to be calibrated relative to the robot Base coordinate system. After expanding equation (1), we can get:

$$\boldsymbol{R}_0 \cdot \boldsymbol{X}_\mathrm{t} + \boldsymbol{T}_0 = \boldsymbol{R}_\mathrm{s} \cdot \boldsymbol{X}_\mathrm{L} + \boldsymbol{T}_\mathrm{s} \tag{2}$$

Control Robot and make sensor restore the same fixed point twice, we can get:

$$\boldsymbol{R}_{01} \cdot \boldsymbol{X}_\mathrm{t} + \boldsymbol{T}_{01} = \boldsymbol{R}_\mathrm{s} \cdot \boldsymbol{X}_\mathrm{L1} + \boldsymbol{T}_\mathrm{s} \tag{3}$$

$$\boldsymbol{R}_{02} \cdot \boldsymbol{X}_\mathrm{t} + \boldsymbol{T}_{02} = \boldsymbol{R}_\mathrm{s} \cdot \boldsymbol{X}_\mathrm{L2} + \boldsymbol{T}_\mathrm{s} \tag{4}$$

Make the robot remain the same posture in the control process, i.e.$R_{01} = R_{02}$, the following equation can be obtained by the equation (2) to (4):

$$\boldsymbol{T}_{02} - \boldsymbol{T}_{01} = \boldsymbol{R}_\mathrm{s} \cdot \left(\boldsymbol{X}_\mathrm{L2} - \boldsymbol{X}_\mathrm{L1} \right) \tag{5}$$

After collecting a plurality of sets of experimental data, we can obtain R_s by solving equation (5). But for the line-structured light sensor, searching for the coordinates of a space point under sensor is difficult. You can solve this problem by allowing the sensor to restore the virtual space points (such as the center of a ball). When the sensor scanning the ball, which laser line can fit a space round. Then the center of the ball can be determined by the geometry of the case of the ball radius sphere is known. At this time, there will be two solutions for the center of balls. They are electing upper, middle and lower three-point of the scan line through the agreed order, and get the cross product based on these three points to two vector quantities can determine the general direction of where the center of the ball. Thus we can remove the pseudo-solution [8].

2) To solve the calibration parameters Ts by the method of scanning ball surface

By the equation (2), for the space points that have the fixed position with $Tool_0$ satisfy the following formula:

$$\boldsymbol{X}_{\mathrm{t}} = \boldsymbol{R}_0^{-1} \cdot \left(\boldsymbol{R}_{\mathrm{s}} \cdot \boldsymbol{X}_{\mathrm{L}} + \boldsymbol{T}_{\mathrm{s}} - \boldsymbol{T}_0 \right) = \boldsymbol{R}_0^{-1} \cdot \left(\boldsymbol{R}_{\mathrm{s}} \cdot \boldsymbol{X}_{\mathrm{L}} - \boldsymbol{T}_0 \right) + \boldsymbol{R}_0^{-1} \boldsymbol{T}_{\mathrm{s}} \tag{6}$$

When the robot scanning the surface of a ball by the translational movement (posture unchanged) (show as figure 4), the relationship between the restored result of the sensor X_t (the position of spatial point relative to the $Tool_0$ coordinate system) and it's coordinate X_L under the sensor coordinate system show as equation (6). For the different spatial point X_{t1} and X_{t2}, we can get its relative position relationship:

$$\boldsymbol{X}_{\mathrm{t2}} - \boldsymbol{X}_{\mathrm{t1}} = \boldsymbol{R}_{02}^{-1} \cdot \left(\boldsymbol{R}_{\mathrm{s}} \cdot \boldsymbol{X}_{\mathrm{L2}} + \boldsymbol{T}_{\mathrm{s}} - \boldsymbol{T}_{02} \right) - \boldsymbol{R}_{01}^{-1} \cdot \left(\boldsymbol{R}_{\mathrm{s}} \cdot \boldsymbol{X}_{\mathrm{L1}} + \boldsymbol{T}_{\mathrm{s}} - \boldsymbol{T}_{01} \right) \tag{7}$$

When the robot only do the translational motion during the scanning process, i.e. $R_{01} = R_{02,}$ the above equation can simplify to

$$\boldsymbol{X}_{\mathrm{t2}} - \boldsymbol{X}_{\mathrm{t1}} = \boldsymbol{R}_{01}^{-1} \cdot \left[\boldsymbol{R}_{\mathrm{s}} \cdot \left(\boldsymbol{X}_{\mathrm{L2}} - \boldsymbol{X}_{\mathrm{L1}} \right) \boldsymbol{X}_{\mathrm{L2}} - \boldsymbol{T}_{02} + \boldsymbol{T}_{01} \right] \tag{8}$$

From equation (8), we can get: the relative positional relationship between the recovered results (i.e. the recovered shape of the object) is only linking with the rotation matrix R_s. No matter what value of T_s is, the recovered result is always a sphere. Take $T_s = 0$, make its recovered scan results (i.e. X_t) do spherical fit, the relationship between the obtained position of the center of the ball X_B and the true position of the center of the ball X_b will satisfy the following formula:

$$\boldsymbol{X}_{\mathrm{b}} = \boldsymbol{X}_{\mathrm{B}} + \boldsymbol{R}_0^{-1} \boldsymbol{T}_{\mathrm{s}} \tag{9}$$

We can get multiple groups of X_B and R_0 by changing the posture of the robot to scan the surface of the ball, through solving equation (9) to get Ts.

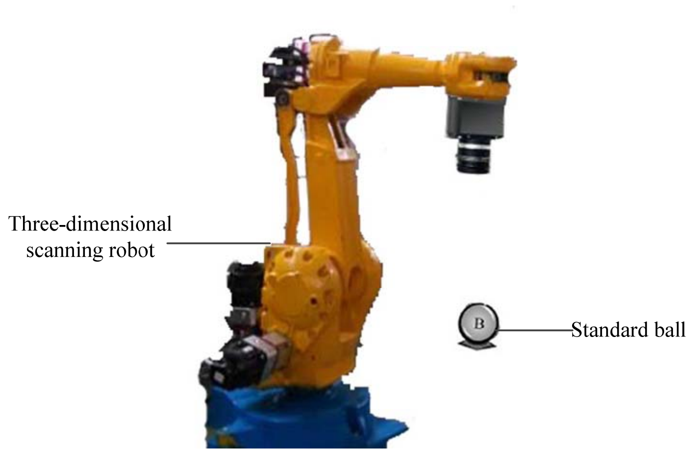


Figure 4. Calibration schematic of three-dimensional scanning robot system

IV. ERROR MODELLING OF THE 3D SCANNING ROBOT SYSTEM

As we known, without considering the effects of noise measurement, the error of the whole system comes mainly from the following several aspects: body positioning error of the robot, the data acquisition error of the line-structured light sensor and the error brought from hand-eye matrix. Whether all aspects of the 3D scanning robot system error are considerable, the last error is due to the obtained measurement data. Extraction of the system overall error is as follows: First, using 3D scanning robot system to scan a standard work piece. After obtaining the 3D discrete point cloud of the work piece, make the obtained point clouds and standard CAD models of the work piece be scanned alignment. Then we search for the nearest point of each discrete point cloud (i.e. measurement points) before the registration on the CAD model. To complete the registration of standard CAD model and point cloud by the iterative closest point (ICP) algorithm of the registration algorithm [9], the nearest point corresponding to the point cloud after registration is considered as the corresponding real point of the point approximately, and then consider the difference between the real point and the measuring point as the error signal of the system.

A. ICP ALGORITHM: ITERATIVE CLOSEST POINT ALGORITHM

Establishing the corresponding point sets P and Q according to certain criteria, the number of corresponding point pairs is n. Before calculation of registration, first of all, is finding the corresponding closest point of point cloud (i.e. the closest point that in line with some geometric constraints conditions.

After finding the closest point, we can make the point cloud registration with the CAD model to the same coordinate system by iteration of least squares method finally. The iterative closest point (ICP) algorithm is calculating the optimal rotation matrix R and translation vector T between point set P and Q, in order to make following error function E reach the minimum.

$$E(\boldsymbol{R},\boldsymbol{T}) = \frac{1}{n}\sum_{i=1}^{n}\left\|\boldsymbol{R}\boldsymbol{P}_i + \boldsymbol{T} - \boldsymbol{Q}_i\right\|^2 \tag{10}$$

In order to not loss of generality, assuming point set P is a subset of the point set Q, the closest points of each point in Q can be found in P to constitute the point set Q '. The number of points in Q' are same as the number of points in Q. Then we can solve the rotation matrix R and translation vector T between Q 'and Q by optimization method. The goal is making the distance between corresponding points of Q' and Q reach minimum, i.e. The result of equation (10) is the minimum.

In fact, it is difficult to obtain the true points corresponding to measurement point precisely, in order to get the real point of the standard work piece corresponding to measurement points, the practice of this article is fixing the obtained measuring points and then using the ICP registration algorithm do the registration of CAD model of the standard work piece and point cloud, considering the closest point corresponding to point cloud after registration as its real point.

B. RBFNN: RADIAL BASIS FUNCTION NEURAL NETWORK

Radial basis function neural networks (RBFNN) were proposed by Broomhead and Lowe (1988). Radial basis function neural network is the neural network based on multivariable interpolated radial basis function. It is able to approach continuous functions by arbitrary precision [10].The basic architecture of a RBF neural network is shown in figure 5. The input is the coordinates (u, v) in the image coordinate system. The output is the error of the corresponding measuring point. The RBF neural network has three layers. The first layer is the input layer which accepts the two dimensional input data (u, v). The second layer is called hidden layer, which is composed of radial basis function neurons that achieve a non-linear mapping. The third layer is the output layer, which is a linear combination of each hidden layer neuron output. The expected error in the output layer is obtained by the ICP algorithm.

Overall, network mapping from the input to the output is non-linear. But the network output tunable is linear. So each training samples require only a small amount of adjustment of weights and thresholds. Then we can avoid falling into the local minimum point [11].

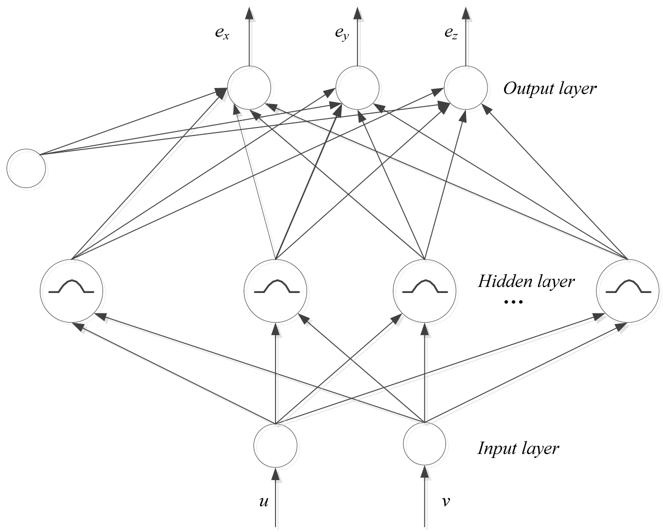


Figure 5. The REFNN for error modeling of 3D scanning robot system

Choose Gaussian function as the hidden layer neurons function, its radial is symmetry and smooth, the expression is as follows:

$$\phi_i\left(\boldsymbol{X}\right) = \exp\left(\frac{-\left\|\boldsymbol{X}-\boldsymbol{u}_i\right\|^2}{\sigma_i^2}\right) \tag{11}$$

The output of the RBF neural network is computed according to equation (12):

$$y = f\left(x\right) = \sum_{i=1}^{N} w_i\phi_i\left(\boldsymbol{X}\right) + w_0 \tag{12}$$

Where $X=(x_1,x_2,\ldots,x_m)^T \in R^{m\times 1}$ is an input vector. $u_i=(u_{i1},u_{i2},\ldots,u_{im})^T \in R^{m\times 1}$ is the neurons center vector, and σ_i is the corresponding width of the i-th neurons. N is the number of neurons in the hidden layer, while $\|\square\|$ denotes the Euclidean distance. $w= (w_1, w_2 ,\ldots,w_N)^T$ is the vector of weights between the hidden layer and the output layer .w_0 is the bias which allows the sensitivity of the radial basis function neuron to be adjusted.

C. PSO-RBFNN: PARTICLE SWARM OPTIMIZATION RADIAL BASIS FUNCTION NEURAL NETWORK

Particle swarm optimization (PSO)algorithm is the global optimization techniques based on swarm intelligence，that is proposed by Eberhart RC and Kennedy J. (1995) [12].In this algorithm, a candidate solution is presented as a particle, which explores the searching place to find the global solution [13]. The basic PSO algorithm was developed to solve optimization problems with continuous-valued parameters. Each particle has a position x in the search space and a velocity v indicating direction and step-size of change in the current position. Each particle keeps track of the quality of the solution to the optimization problem it represents[14].Let i be a particle in an N-dimensional search space with velocity $v_{i,j}$, position $x_{i,j}$, personal best position $p_{i,j}$, and the best position visited in the past by a particle in its neighborhood $p_{g,j}$. The velocity updates formula of i-th particle in the j-dimensional as follows:

$$v_{i,j}(t+1)=\omega v_{i,j}(t)+c_1r_1\left[p_{i,j}(t)-x_{i,j}(t)\right]+c_2r_2\left(p_{g,j}(t)-x_{i,j}(t)\right)$$
$$i=1,2,\ldots,m \qquad j=1,2,\ldots,n \tag{13}$$

Equation (13) computes the magnitude of change in the particle's position in each dimension j. The t denotes the iterations number of current group. The ω is called the inertia weight, determines the acceleration or deceleration in the current direction. The c_1 is the cognitive component weight. The c_2 is the social component weight. The r_1 and r_2 are n-dimensional random vectors with each $r_{1,j}$, $r_{2,j}$ ~ U(0, 1) drawn independently. The n is the particle dimension. The m is the particle number of the group. The position is updated by adding the updated velocity to the current position:

$$x_{i,j}(t+1)=x_{i,j}(t)+v_{i,j}(t+1) \tag{14}$$

In this paper, using PSO to optimize the center and width values of RBFNN neurons and choose the sum of square of error between actual output and the desired output as the optimization objective function. The error signal is the desired output of the system, which is obtained by calculation of the ICP algorithm. The position of each particle is corresponding to the center of the width of the neurons in the parameter space. The initial position of the center of the neurons is determined by the fuzzy c-mean method. The best particle represents the minimum value of the objective function. The entire calculation process of PSO is shown in figure 6.

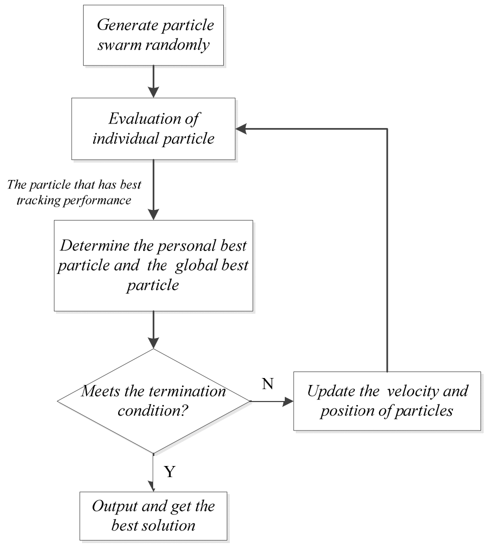


Figure 6. Flowchart of PSO algorithm

In PSO algorithm, constants c_1, c_2 represents the stochastic acceleration term weight that makes the particles move toward to the particle that has the best fitness. If its value is small, the convergence rate is low. Conversely, if its value is too big, there may be beyond the best position, causing the instability of the algorithm [15]. According to the above two kinds of experience, so we make the acceleration factor constant $c_1 = c_2 = 2.0$. Meanwhile, controlling the value of inertia weight can regulate global search and local search optimization capability of the algorithm. When the value of ω is large, global search capability is strong but local search capability is weak. On the contrary, the local search ability is strong but the global search capability is weak. In this paper, taking the method of inertia weight ω decreases linearly from 0.9 to 0.4.Its values are taking according to the following equation:

$$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{t_{\max}} \times t \tag{15}$$

In this equation, the t_{max} represents the maximum number of iterations. The t represents the current number of iterations. Meanwhile, choosing the mean square error（MSE）of the network output as the adaptive function is expressed simply as:

$$E = \frac{1}{N}\sum_{i=1}^{N}\left[y(k) - \hat{y}(k)\right]^2 \tag{16}$$

They represent the desired output and $\hat{y}$ represent the output of the network. Meanwhile, selecting the size particle group is 20.Assuming the maximum number of iterations is 100 by the method of trial and error.

Once the parameters of the RBF hidden layer neurons (namely the centers of the neurons and the Gaussian function widths) are determined, the other is the calculation of the connection weights between the hidden and output layers. The output weights between the hidden and output layers are directly calculated by the least squares pseudo inverse.

$$\boldsymbol{w} = \left(\boldsymbol{H}^T \boldsymbol{H}\right)^{-1} \boldsymbol{H}^T \boldsymbol{y} \tag{17}$$

$$H = \begin{bmatrix} \phi_1(X_1) & \phi_2(X_1) & \cdots & \phi_N(X_1) & 1 \\ \phi_1(X_2) & \phi_2(X_2) & \cdots & \phi_N(X_2) & 1 \\ \vdots & \vdots & & \vdots & \vdots \\ \phi_1(X_M) & \phi_2(X_M) & \cdots & \phi_N(X_M) & 1 \end{bmatrix}$$

V EXPERIMENTAL ANALYSIS

The key of using PSO-RBFNN to complete system error modeling is how to construct the input and output data pairs [16]. The coordinates of the image coordinate system is selected as the input, namely (u, v).Since the size of the image coordinate system is 576 × 768, we adjust the robot to the different positions repeatedly to get enough training data pairs. The process of structuring input and output data pairs is given in figure 7.In this method, we must try to make the obtained input data (i.e., the image coordinates) fill the entire image plane as far as possible. After obtaining the input and output data pairs, we divide the data into training data and test data. The training data is used for learning of the PSO-RBFNN. Test data is used for testing the effect of the training network. The specific process is shown in figure 8.

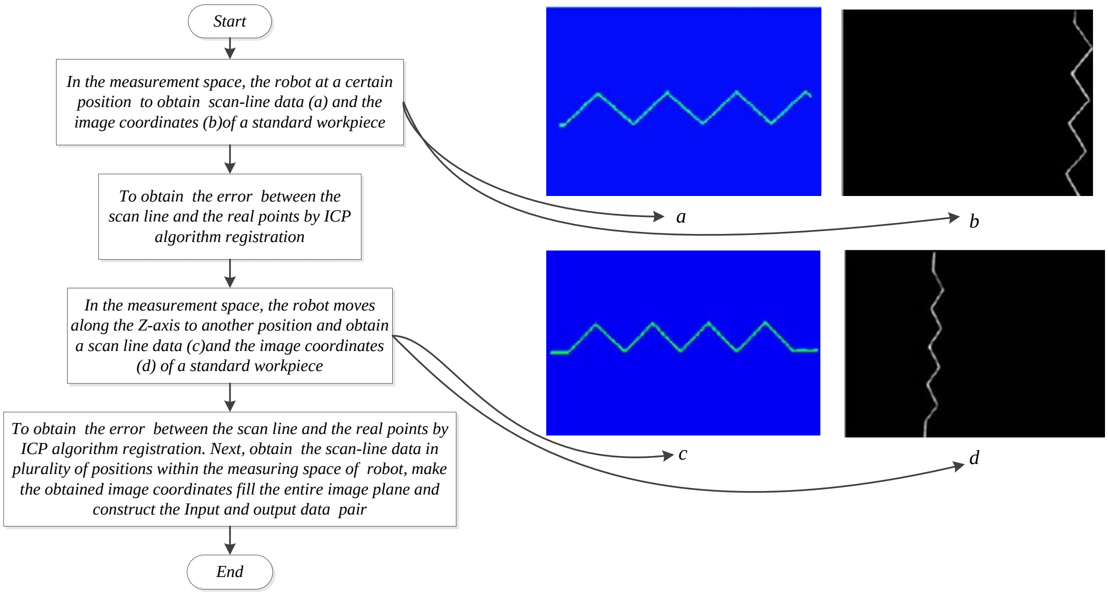


Figure 7. Input and output data structure of system error modeling

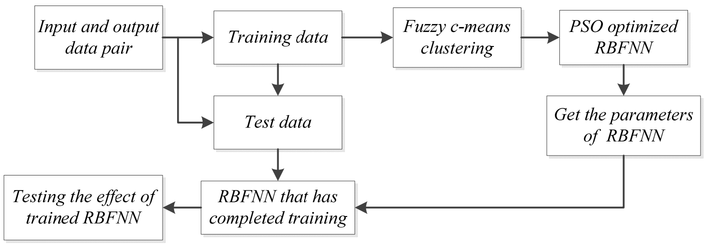


Figure 8. PSO-RBFNN training process for system error modeling

According to the error modeling process shown in Figure 7, we establish the error network model of 3D scanning robot system. Firstly, through standard work piece that is used for establishing the network model and the other standard work piece that not involved in the modeling to validate the correctness of the resulting network. Next we verify the validity and effectiveness of the proposed error modeling and compensation method by measuring the distance between the centers of two standard balls.

A. TO VERIFY THE CORRECTNESS OF THE ERROR MODEL

According to the error modeling process shown in figure 7, in the sensor measurement space, we choose standard work piece I(figure.10 (a)) to measure. The robot gets a scan line and the its three-dimensional coordinate data in the image coordinate system along the z-axis direction at intervals of 2mm, obtaining 24 sets of scan-line data and the corresponding three-dimensional data error altogether. Each scanning line is corresponding to 576 data points. So the data amount is $24 \times 576 = 13824$. Obtained input and output data is divided into two parts. One is used for training the network. The other is used for testing network performance. By using RBFNN we can establish a network model. At the same time, in order to verify the effect of particle swarm compressing the scale of the trained RBFNN. Then we use PSO algorithm to optimize the neurons center and its width of RBFNN. Comparison results of the network established by the two methods are shown in table 1.

Table 1: Training result comparison between RBFNN and PSO-RBFNN

Algorithm category	Network structure	Training time/s	Training error	Test error
RBFNN	2-100-3	5.32	1.73×10^{-5}	4.40×10^{-5}
PSO-RBFNN	2-80-3	4.88	1.7569×10^{-4}	2.63×10^{-5}

We can be seen from table 1: When using PSO algorithm for training RBFNN, there is no advantage in the training time comparing to the RBFNN. But the number of the hidden layer neurons of the neural network reduces almost in the same training time, compressing the size of the RBFNN network to some extent. Meanwhile, the training error E reduces an order of magnitude. In addition, the test error of the PSO-RBFNN is significantly reduced, indicating the generalization ability of the network model getting a relatively large increase. Figure 9 is the convergence comparison of the two neural network algorithms. We can see that PSO-RBFNN network model able to achieve a high training accuracy in fewer iterations t.

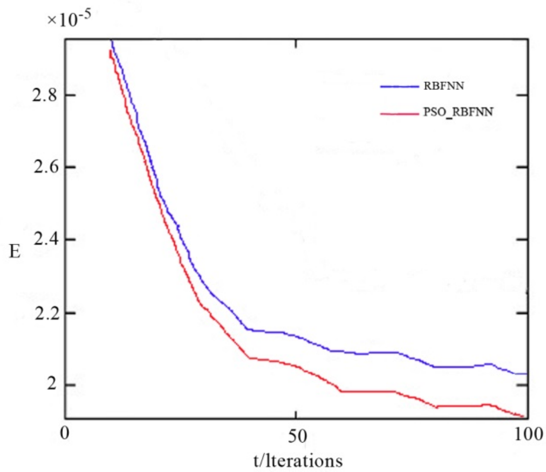
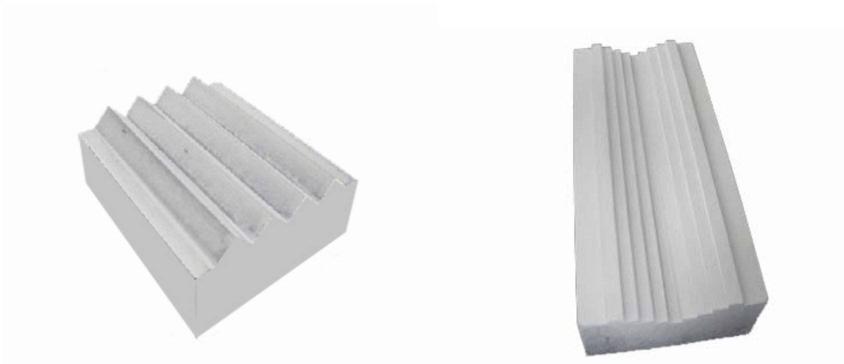


Figure 9. Convergence comparison between RBFNN and PSO-RBFNN

After establishing error network model, we control the robot to scan the standard work piece I (figure.10 (a)) and standard work piece II (figure.10 (b)) that not involved in modeling within the measurement space. Then it conducts the error compensation to the obtained three-dimensional data.



(a) Standard work piece Ⅰ (b) Standard work piece II

Figure10. Pictures of standard work piece

After the measurement, we make the data after the compensation and the data before the compensation conduct registration with the corresponding standard CAD model. The error before the compensation and after the compensation is shown in table 2. As we can see, using the proposed error network model to conduct the error compensation can reduce measurement errors and improve the measurement accuracy.

Table 2: The error comparison of standard between before compensation and after compensation /mm

Standard workpiece	Maximum positive error		Maximum negative error		Average error	
	Before compensation	After compensation	Before compensation	After compensation	Before compensation	After compensation
Ⅰ	0.164	0.137	-0.217	-0.084	0.028	0.016
II	0.215	0.193	-0.242	-0.137	0.033	0.025

B.TO VERIFY THE VALIDITY OF THE ERROR COMPENSATION METHOD

Two standard balls A and B are fixed on a plane, taking the average of 10 times distances between the centers of the two balls measured by coordinate measuring machine (CMM) as the true value of the distance between the centers of the two balls. We can control the robot to scan two standard balls (figure 11) within the measurement space, and conduct the error compensation to the obtained spherical three-dimensional data. We get the ball center coordinates and radius value by fitting the data after the compensation and the data before the compensation. The experimental results are shown in table3.

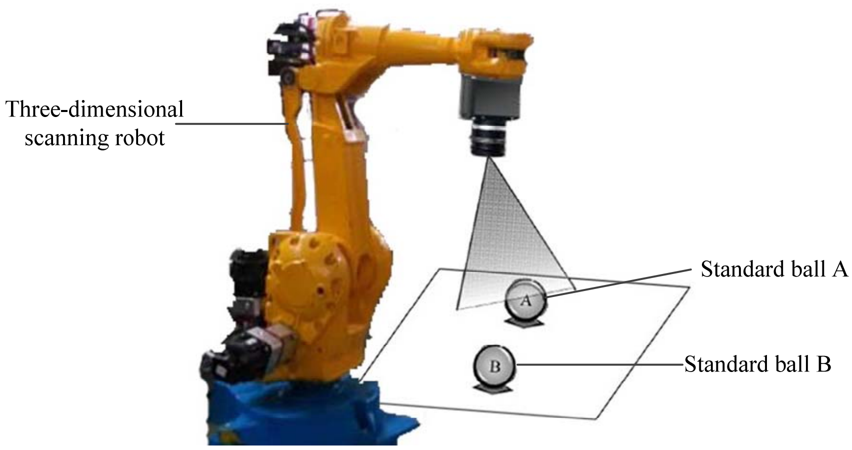


Figure 11.Schematic diagram of error compensation experiment for three-dimensional scanning robot

Table 3: Measuring results of the distance between the centers of two goals /mm

No.	Measured distance by CMM	Obtained distance before compensation	Obtained distance after compensation
1	599.7622	599.7710	599.7634
2	599.7603	599.8523	599.8021
3	599.7577	599.7642	599.7623
4	599.7631	599.8124	599.7831
5	599.7620	599.7847	599.7802
6	599.7646	599.7728	599.7653
7	599.7662	599.8701	599.8217
8	599.7599	599.7831	599.7662
9	599.7628	599.8252	599.7815
10	599.7605	599.7815	599.7668
Average	599.7619	599.8017	599.7793

From the measurement results shown in table 3, we can get that: in terms of the average distance between the centers of two balls, the error after compensation reduces 0.0224mm than the error before the compensation. Meanwhile, the maximum error between the centers of two balls from 0.1082mm before compensation is reducing to 0.0598mm. For the No.2, 4, 7, 9 measurements that have large error, the error after compensation decreases 0.03 ~ 0.05mm. However, in case of the measurement accuracy being already high, such as No.1, 3, 5, 6,8,10 measurements, the error after compensation is also lower 0.002 ~ 0.015mm than the error before compensation. The measurement accuracy is enhanced. Thus we verify the correctness and effectiveness of the proposed error modeling and compensation method.

Figure 12. Experimental platform

VI CONCLUSIONS

This paper presents a method to establish the error modelling of 3D scanning robot system by using particle swarm optimization radial basis function neural network (PSO-RBFNN). The methods use the ICP algorithm to construct the input and output data pairs of the neural network cleverly. PSO-RBFNN compresses the scale of the network and reduces the training error to some extent under the premise of ensuring network performance. We can establish the error model by scanning the standard work piece, and then through measuring the distance between two standard ball centers, verify the correctness and effectiveness of the proposed error modeling and compensation method. Improving the measurement accuracy of the robot is a complicated process. The error model and compensation method presented in this paper effectively reduces the measurement error, but still needs to continue to study in order to meet higher accuracy demands on the actual measurement in the later period.

ACKNOWLEDGEMENT

This work is supported by Project supported by the 2012 National Major Scientific Instruments Development Special Foundation (No.2011YQ150040) and the 2011 Shanghai Major Technology and Equipment Development Special Foundation （ZB-ZBYZ-03-11-1709）,and the Innovation Fund Project For Graduate Student of Shanghai

（JWCXSL1302）

REFERENCES

[1] Larsson SOREN and KJELLANDER J.A.P, “An industrial robot and a laser scanner as a flexible solution towards an automatic system for reverse engineering of unknown objects,” Proceedings of the 7th Biennial Conference on Engineering Systems Design and Analysis ,pp.341-350, 2004.

[2] Li Jianfeng,Zhu Jianhua and Tang Qing,“Scanning method of robot-3D scanner system and its industrial application,”Opto-Electronic Engineering, Vol. 34, No. 2, pp. 15-21, February 2007.

[3] Jianhao Zhang and Jinda Cai, “Error Analysis and Compensation Method Of 6-axis Industrial Robot,” International Journal on Smart Sensing and Intelligent System, Vol.6,No.4, September 2013, pp. 1383-1399.

[4] MA Zi，XU Huipu，HU Ying and et al,“A novel robot surface measurement system enhanced with 3D surface reconstruction,” International Journal of Modeling Identification and Control，Vol. 4, No. 3, pp. 286-298, 2008.

[5] CHAUMETTE F and HUTCHINSON S,“Visual servo control I: basic approaches,” IEEE Robotics and Automation Magazine, Vol.13, No.4, pp.82-90, 2006.

[6] Wu Defeng and Li Aiguo,“Error modeling and compensation approach for three-dimensional surface scanning robot,”Journal of Mechanical Engineering, Vol.48,No.13,pp.61-67, 2012.

[7] Peter BIBER，Henrik ANDREASSON，Tom DUCKETT and et al,“3D modeling of indoor environments by a mobile robot with a laser scanner and panoramic camera,”2004 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS),Sendai, Japan : IEEE,pp.3430-3435, April 2004.

[8] Biqiang Du , Ning Xi , and Erick Nieves,“Industrial robot calibration using a virtual linear constraint,” International Journal on Smart Sensing and Intelligent System, Vol.5,No.4, December 2012, pp. 987-1001.

[9] Ziegert J, Datseris p. “Basic Consideration for Robot Calibration,” IEEE International

Conference on Robotics and Automation, Philadelphia, 1988, pp.932-938.

[10] You Wenjian,Liang Bing and Li Yinjun,"Reserch of output characteristic fitting of eddy-current sensor based on radial-basis function neural network,"Industry and Mine Automatic, Vol.39,No.2,pp.47-50, February 2013.

[11] MEHRABI S，MAGHSOUDLOO M，ARABALIBEIK H and et al,"Application of multilayer perceptron and radial basis function neural networks in differentiating between chronic obstructive pulmonary and congestive heart failure diseases,"Expert Systems with Applications,Vol.36, No.3, pp. 6956-6959, 2009.

[12] SENTH IL BRUMMAGEM M, RAOM V C and CHANDRAMOHAN A,"A new and improved version of particle swarm optimization algorithm with global-l local best parameters,"Knowledge and Information Systems, Vol.16, No.3, pp.15-26, 2008.

[13] Mehdi Neshat, Mehdi Sargolzaei, Azra Masoumi and et al,"A new kind of PSO: predator particle swarm optimization," International Journal on Smart Sensing and Intelligent System, Vol.5,No.2, June 2012, pp. 521-539.

[14] Joost Langeveld , Andries P. Engelbrecht,"Set-based particle swarm optimization applied to the multidimensional knapsack problem," Swarm Intelligence,Vol.6,No.4,pp.297-342,2012.

[15] B ERGH F and ENGELBRECHT A P,"A cooperative approach to particle swarm optimization,"IEEE Trans on Evolutionary Computation, Vol.8, No.3, pp.1-15, 2004.

[16] DEL V，VENAYAGAMOORTHY G，MOHAGHEGHIS and et al, "Particle swarm optimization：Basic concepts，variants and applications in power system,"IEEE Transactions on Evolutionary Computation，Vol.12, No.2, pp.171-195, 2008.