Hindawi

Retraction

Retracted: Industrial Robot Assembly Line Design Using Machine Vision

Journal of Robotics

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope

(2) Discrepancies in the description of the research reported

(3) Discrepancies between the availability of data and the research described

(4) Inappropriate citations

(5) Incoherent, meaningless and/or irrelevant content included in the article

(6) Manipulated or compromised peer review

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] X. Xie, "Industrial Robot Assembly Line Design Using Machine Vision," *Journal of Robotics*, vol. 2023, Article ID 4409033, 13 pages, 2023.

Research Article
Industrial Robot Assembly Line Design Using Machine Vision

Xiang Xie [iD]

Chongqing Creation Vocational College, Yongchuan 402160, China

Correspondence should be addressed to Xiang Xie; xiexiang202207@126.com

In order to further improve the functional requirements and performance indicators of the industrial robot assembly system and more accurately realize the measurement and recognition of the target position of the assembly line by the vision system, this article constructs a robot assembly line system based on obstacle detection and robot arm obstacle path planning based on machine vision technology and further improves the intelligence and accuracy of the assembly line system through the design and optimization of the system software module. Through the experimental verification of the positioning error based on the eye-to-hand binocular vision system and eye-in-hand monocular vision system, the system proposed in this article meets the design accuracy requirements of less than 0.1 mm in x/y direction and less than 1 mm in depth direction and verifies the feasibility and high accuracy of the system.

1. Introduction

The development of science and technology has not only impacted the traditional manufacturing industry but also provided strong technical support for the production innovation and upgrading of the traditional manufacturing industry. The traditional manufacturing industry gradually began to transform and upgrade towards a more intelligent production line. At this stage, because the production and assembly of reducer requires relatively high precision and most of them need to be assembled manually, the work efficiency is low. The intervention of industrial robots can greatly improve the production efficiency of the assembly line, minimize the investment of traditional labor, and further improve the labor efficiency of the assembly line and the comprehensive economic benefits of the enterprise. The industrial robot assembly line is mainly through the combination of industrial robots, machine vision, and automatic assembly equipment to jointly complete the processes of mass product storage, transportation, detection and assembly, which further improves the efficiency of traditional production and manufacturing. Based on this, this article discusses and designs a higher precision assembly line system from the direction of improving and perfecting the functional requirements and performance indicators of the industrial robot assembly system [1, 2].

2. Literature Review

Taking the assembly process of automobile gearbox as the research goal, this article studies the key technologies that need to be solved in the automatic assembly line and provides a theoretical reference for the actual gearbox assembly work. According to the requirements of the main engine factory and the purpose of adapting to the batch production of single variety and realizing the replacement of machines, an industrial robot air filter assembly line has been developed and designed by the passenger car assembly production line of the No. 2 air filter manufacturing plant. This production line is the key equipment for the production of the passenger car air filter. A company has developed a fully automatic assembly line of automobile interior door handles based on industrial robots. This assembly line can reduce labor costs by 90%, increase production efficiency to the original 400%, and reduce the monthly average failure rate to less than 1%. It provides a feasible reference scheme for the upgrading and transformation of manufacturing methods in the same industry [3]. Talaat et al. pointed out the necessity of industrial robots in wind turbine blade assembly and the enlightenment to the establishment of wind turbine blade automatic assembly production line by analyzing the problems existing in the existing wind turbine blade

assembly and combining the application of industrial robots in automobile assembly and aircraft assembly [4]. However, these production lines are not intelligent enough and still need manual intervention. They have not really realized fully automated production. With the development of science and technology, the birth of more efficient fully automated production lines is called for. Kumar et al. deeply studied the kinematics and trajectory interpolation of palletizer, discussed the application of the scattered node interpolation algorithm in robot trajectory planning, and used this algorithm to interpolate the robot path trajectory to make the robot motion trajectory smoother. Through the simulation and comparison of joint space trajectory planning of the industrial robot through cubic polynomial interpolation and quintic polynomial interpolation, the advantages and disadvantages of the two trajectory planning methods are obtained, which has good reference significance for the trajectory planning of the robot [5]. Zhang et al. proposed a smooth curve transition algorithm for continuous trajectory planning of industrial robot operation space to improve the motion speed of industrial robots during continuous trajectory transition. Under the condition of ensuring the accuracy of continuous trajectory transition, the algorithm tries to improve the speed of the transition area, reduce the transition time, and make the calculation process as simple as possible, so as to be used for online planning and improve the versatility of the algorithm [6].

3. Target Workpiece Recognition Based on the BP Neural Network

3.1. Artificial Neural Network.
The artificial neural network is composed of a large number of artificial neurons. Each input x_j in the artificial neuron has a weight w_{ij} related to it. In the processing area, the sum of the input and its corresponding weight values are calculated to calculate the unique output. The mapping relationship between the output y and the weight sum is called the transfer function of neurons. This process can be expressed by the following formula:

$$y = f\left(\sum_j w_{ij} x_j + b\right), \tag{1}$$

where f represents the type of the transfer function used by the neuron in the neural network.

3.2. BP Neural Network Structure Algorithm.
The basic idea of using the BP learning algorithm to train the feedforward network is as follows: First, calculate the actual output value of the input characteristics in the training sample after passing through the feedforward network through a forward process. Then, compare the output value with the expected value to get the error signal and then adjust the connection strength between each neuron from back to front according to the error signal. Repeat the forward process calculation again to reduce the error and then compare the new output value with

the expected value to obtain a new error signal smaller than the original. Then, according to the small error signal, the connection strength between neurons in each neural network layer is adjusted from back to front. This forward process and backward process are carried out repeatedly until the error meets the requirements. In order to determine all the connection weights of the network, these weight parameters are generally learned through the training sample set [7, 8]. A training set containing n samples can be expressed as

$$X = \{x_n, t_n\}_{n=1}^N, \tag{2}$$

where x_n represents the n_i dimensional input characteristics corresponding to the sample and t_n is the n_0 dimensional category output variable.

The define loss function is as follows:

$$l(\Theta; x_n, t_n). \tag{3}$$

To calculate the loss value of a specific set of model parameters on a sample, where Θ is a parameter set, representing all connection weights, namely,

$$\Theta = \{W_{ij}^l, b_j^l\}, \tag{4}$$

where W_{ij}^l refers to the weight connecting the jth node of the layer l and the ith node of layer $l + 1$ and b_j^l refers to the offset term of the ith node of the layer $l + 1$. In order to obtain the optimal connection weight, the following optimization problems need to be solved:

$$\widehat{\Theta} = \underset{\Theta}{\text{argmin}} \frac{1}{N} \sum_{n=1}^N l(\Theta; x_n, t_n). \tag{5}$$

The goal of this optimization problem is to obtain specific parameter values to minimize the total loss of the loss function on the training set. If the square sum error is taken as the optimization objective, the loss function $l(\Theta; x_n, t_n)$ can be expressed as

$$l(\Theta; x_n, t_n) = \frac{1}{2} \|t_n - y_n\|^2, \tag{6}$$

where y_n represents the output function value obtained when the input of the feedforward network is x_n. Accordingly, the target loss function on the whole training set can be expressed as

$$L\Theta = \frac{1}{2} \sum_{n=1}^N l(\Theta; x_n, t_n). \tag{7}$$

The gradient direction of the objective function relative to parameter Θ is

$$\Delta L\Theta = \frac{\partial L\Theta}{\partial \Theta} = \frac{1}{N} \sum_{n=1}^N (h_\Theta(x_n) - t_n) \frac{\partial h_\Theta(x_n)}{\partial \Theta}. \tag{8}$$

Assuming that the initial value of parameter Θ is Θ_0, the parameter Θ is continuously modified according to the gradient descent method.

$$\Theta_{k+1} = \Theta_k + \Delta\Theta_k = \Theta_k - \eta \Delta L(\Theta_k), \tag{9}$$

where

$$\Delta\Theta_k = -\eta\Delta L(\Theta_k). \tag{10}$$

The above formula represents the update amount of the kth iteration parameters and $\eta \in (0, 1)$ is the learning rate.

Assuming that the nonlinear function used by the computing nodes in the network is the sigmoid function, in order to facilitate the later derivation of the back-propagation algorithm, the total input of the ith node of the lth layer of the feedforward network is recorded as z_i^l and its output is recorded as a_i^l. For the hidden layer of the network, there are

$$\begin{cases} a_i^l = \sigma(z_i^l) = \dfrac{1}{1 + \exp(-z_i^l)}, \\ z_i^{l-1} = \displaystyle\sum_{j=1} W_{ij}^{l-1} a_i^{l-1} + b_i^{l-1}. \end{cases} \tag{11}$$

Assuming that the error of the current network is $E = L(\Theta)$, for the connection weight ΔW_{ij}^l and offset Δb_i^l, the update amount in an update process is, respectively,

$$\begin{cases} \Delta W_{ij}^{n_l} = \dfrac{\partial E}{\partial W_{ij}^{n_l}}, \\ \Delta b_i^j = \dfrac{\partial E}{\partial b_i^j}. \end{cases} \tag{12}$$

The back-propagation process starts from the last layer, continuously applies the chain derivation rule, calculates ΔW_{ij}^l and offset Δb_i^l layer by layer from back to front, and then updates the parameters of each layer. For the output layer, the update amount of the connection offset $\Delta b_i^{n_l-1}$ is

$$\Delta b_i^{n_l-1} = \frac{\partial E}{\partial a_i^{n_l}}\frac{\partial a_i^{n_l}}{\partial z_i^{n_l}}\frac{\partial z_i^{n_l}}{\partial b_i^{n_l-1}} = -(t_i - a_i^{n_l})\sigma'(z_i^{n_l}). \tag{13}$$

The update amount of its connection weight $\Delta W_{ij}^{n_l-1}$ is

$$\Delta W_{ij}^{n_l-1} = \frac{\partial E}{\partial z_i^{n_l}}\frac{\partial z_i^{n_l}}{\partial W_{ij}^{n_l-1}} = \Delta b_i^{n_l-1} a_i^{n_l-1}. \tag{14}$$

For the hidden layer, its connection offset $\Delta b_i^{(l-1)}$ is updated as

$$\Delta b_i^{(l-1)} = \sum_{j=1}^{s_{l+1}} \frac{\partial E}{\partial z_j^{l+1}}\frac{\partial z_j^{l+1}}{\partial a_i^l}\frac{\partial a_i^l}{\partial z_i^l}\frac{\partial z_i^l}{\partial W_{ij}^{l-1}} = \Delta b_i^{l+1} a_i^{l-1}. \tag{15}$$

3.3. Target Recognition Process Based on the BP Neural Network. Continuous training of the neural network with a large number of data samples is the premise of using the BP neural network to recognize different targets. The samples include the training input and expected output. The training input data is often some feature quantity of the image, and the expected output can be set as the code corresponding to the image. In the process of training neural network with

samples, there are many interference information in the image, which increases the difficulty of network training. Therefore, this article adopts the strategy based on moment invariants to filter out the interference items. The method based on moment invariants combines the moment invariants in the target image and the code of the sorting target in the image to form a training sample, which can change the target image into a set of moment invariants containing 7 numbers. This strategy can simplify the training sample, filter out the interference information, greatly improve the training efficiency, and improve the recognition accuracy at the same time. In order to filter the interference information to the greatest extent, it is also necessary to binarize the target image and make necessary preparations for extracting moment invariant features [9, 10].

In this article, the BP neural network is used to recognize the workpiece, and samples based on moment invariants are used to train the network. The realization process of the recognition algorithm is as follows:

3.3.1. Obtain Target Image Information. Obtain the image of each angle of the object to be recognized.

3.3.2. Image Preprocessing. Graying and binarization form images suitable for moment invariants.

3.3.3. Find Invariant Moment. Highly concentrated image features are used as the basis for image recognition.

3.3.4. Generate Training Samples and Recognition Samples of the Neural Network. After preprocessing the target image taken from multiple angles, the invariant moment of each image is obtained, and the formed vector is combined with the target code to form a training sample. Each image can get a training sample. If the target code in the training sample is removed, the remaining moment invariant sequence can be used as the sample to be identified.

3.3.5. Establish Neural Network. In this article, the BP neural network with the single hidden layer structure is used to recognize the workpiece. Its structure is shown in Figure 1. The transfer functions of 20 neurons in its hidden layer are *s* functions. Since there are three kinds of workpieces to be identified, the output layer is set to three neurons and the transfer function is linear [11].

3.3.6. Training Neural Network. In order to make the training algorithm have the local convergence of the Gauss–Newton method and the global characteristics of the gradient descent method, the Levenberg–Marquardt algorithm with stable performance is used in this article.

3.3.7. Identify the Target. The weight value of the neural network trained by a large number of moment invariant feature samples tends to be stable. When the moment
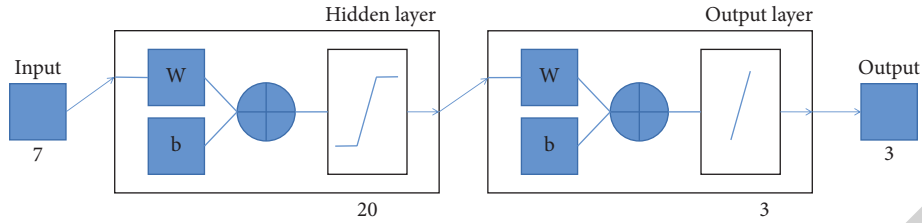
FIGURE 1: Single hidden layer BP neural network structure.

invariant of the image to be recognized is given, the neural network gives the recognition result through mapping.

3.4. Target Workpiece Recognition Experiment. Take 400 pictures of each workpiece from different angles, each picture can extract a set of moment features, and combined with the workpiece label as a group of training samples, a total of 1200 groups of training samples are obtained to train the neural network. In the identification part, 33 pictures of each workpiece were collected, and a total of 99 groups of moment invariants were used as test samples. Through the above training samples, the weights of neurons in the neural network are determined, and the learning of the whole neural network is completed [12]. Then, take the moment invariant feature of the image to be recognized as the recognition data, and the recognition results obtained through the classification of the neural network are drawn as follows. The recognition results of the classifier based on the BP neural network are shown in Table 1.

The total number of recognition errors and the corresponding classification error rate of the 10 weak classifiers constituting the strong classifier are shown in Table 2, and the number of recognition errors and the corresponding error rate of the strong classifier are shown in Table 3.

Comparing and analyzing the data in Tables 2 and 3, it can be found that the classification error rate of strong classifiers is lower than the average classification error rate of 10 weak classifiers, which shows that the recognition effect of multiple neural networks through weighted voting is better than that of a single neural network [13, 14]. The reason why the recognition effect of strong classifier is better is that strong classifier can adjust the weight according to the training situation of each network, constantly weaken useless features, and finally make the overall recognition rate higher.

Comparison of the accuracy of strong and weak classifiers in identifying workpieces is shown in Table 4.

Through a large number of experiments, it can be found that the classifier based on the BP neural network has a high recognition rate in identifying the mechanical workpiece in this article. However, if the angle of shooting the target workpiece pattern is special, it will have a great impact on the recognition effect of the weak classifier. Compared with the strong classifier based on the AdaBoost algorithm, it is more reliable in both the accuracy of recognition and the anti-interference degree of recognition.

TABLE 1: Statistics of recognition results of a single weak classifier.

Error type	Small bolt	Large bolt	Nut	Total error	Error rate (%)
Error statistics	3	3	1	7	7.07

TABLE 2: Statistics of recognition results of multiple weak classifiers.

Weak classifier	1	2	3	4	5	6	7	8	9	10	Total error	Overall classification error rate of weak classifier (%)
Number of errors	6	9	7	8	7	6	9	7	5	9	73	7.37

TABLE 3: Statistics of strong classifier recognition results.

Error type	Small bolt	Large bolt	Nut	Total error	Error rate (%)
Error statistics	2	1	1	4	4.04

TABLE 4: Comparison of recognition effects between strong and weak classifiers.

Classification method	Weak classifier based on the BP neural network (%)	Strong classifier based on the AdaBoost algorithm (%)
Accuracy rate	91.93	94.82

4. Software Design and Implementation of the Robot Vision Assembly Line System

4.1. Software Design Scheme of the Robot Visual Assembly System

4.1.1. Software Requirements Analysis. Robot vision assembly system software is an important part of the system. Its purpose is to realize the integration of the vision system and robot control system [15]. This article adopts the upper computer software and the lower control software to form the software control scheme. The functional block diagram of the system software is shown in Figure 2.

4.1.2. Overall Software Architecture. The software design of this article includes the design of upper computer software and lower control software. The main functions of the upper
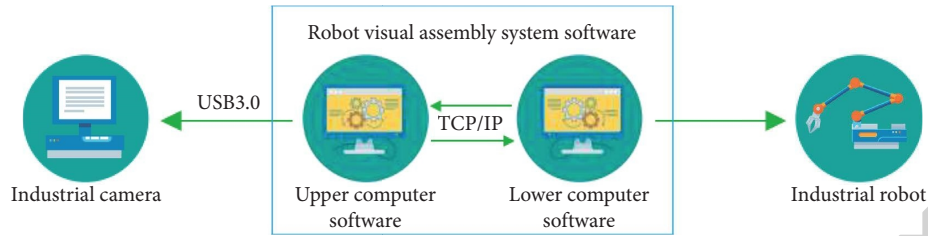
FIGURE 2: System software function block diagram.

computer software include camera image acquisition, visual processing, obstacle visual detection, and data communication. As an important part of the system software, the lower control software mainly realizes the motion control of the manipulator according to the application request of the upper software, so as to achieve the final assembly execution operation. According to the module division, the upper computer software mainly includes the server side of the visual processing module and the data communication module. The lower control software mainly includes the robot motion control module and the client of the data communication module. The overall architecture of the system software is shown in Figure 3.

After the upper software obtains the parameter information of targets and obstacles in the working environment through the visual processing module, it sends the coordinate point information to the lower software through the data communication module through the path planning. After receiving the information, the lower software converts the coordinate point information into position variables and realizes the control of the manipulator through the motion control program.

4.2. Software Modular Design and Implementation

4.2.1. Visual Processing Module. In this article, the eye-to-hand binocular vision system combined with the eye-in-hand monocular vision system is used to locate the target workpiece. Therefore, the whole system shares three cameras. When collecting images, it should be noted that the two cameras fixed on the bracket should be synchronized [16, 17]. For the camera fixed at the end of the manipulator, image acquisition is carried out by calling OpenCV library functions in the visual studio 2010 development environment. The two cameras fixed on the bracket are mercury series cameras developed by Daheng imaging. The way of image acquisition is more troublesome. Because the camera comes with a sealed SDK, when opening and closing the camera, you need to call the SDK of the camera through OpenCV. The overall workflow of the camera is shown in Figure 4.

The main purpose of the software design of the visual positioning function is to package the visual positioning algorithm studied in Chapter 3 with MFC through software development. In this article, the internal and external parameters of the camera are calibrated through the MATLAB camera calibration toolbox. Combined with these parameters, the software design of the visual positioning function is realized through C++ programming and calling the OpenCV computer vision library function. According to the visual localization algorithm studied in the third chapter, the visual localization software design of this article mainly includes the software design of the eye-to-hand binocular vision system and eye-in-hand monocular vision system [18].

4.2.2. Data Communication Module. The data communication module mainly realizes the communication between the upper computer software and the lower control software. The system adopts the TCP/IP network communication protocol to realize the data transmission between the upper computer and the robot controller. The robot controller is used as the client, and the assembly system software is used as the server. The data transmission is carried out by the sending socket. Figure 5 shows the TCP/IP protocol communication flowchart.

4.2.3. Manipulator Motion Control Module. As an important part of the whole system software, the software development of the manipulator motion control module is the final execution part of the whole assembly process. The traditional industrial robot assembly mode realizes the motion control of the manipulator through teaching programming, that is, the robot teaching device programming is used to record the motion trajectory of the manipulator and reproduce these trajectories repeatedly. This article uses visual guidance to realize the motion control of the manipulator, mainly through the lower computer to receive the coordinate information of the upper computer to control the movement of the manipulator and perform the assembly task.

After the vision system recognizes and detects the target workpiece and obstacles in the working environment and obtains their positions, it plans a path without obstacle collision through the obstacle avoidance path planning algorithm. These paths are composed of multiple smaller straight-line motion tracks. According to the step length set by the artificial potential field method, the size of each straight-line segment can be controlled. It will be sent to the manipulator in the form of coordinate point information and stored in the string variable of the manipulator. Therefore, it is necessary to convert these coordinate point information into the motion trajectory of the manipulator through the lower level program. The design flowchart of the manipulator motion control program is shown in Figure 6.
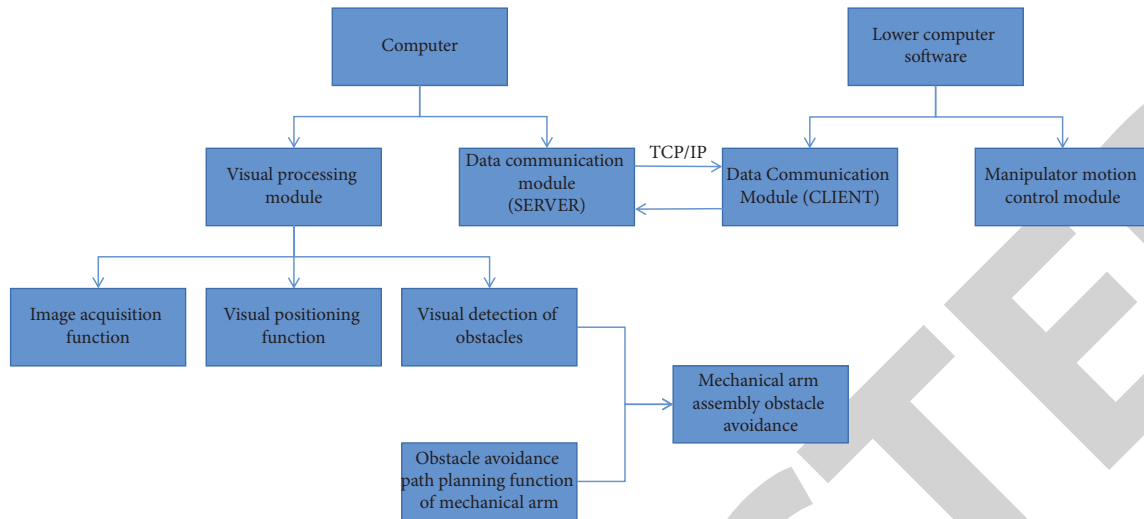
Figure 3: Software architecture of the robot visual assembly system.

4.2.4. Industrial Robot Assembly Program

(1) Trajectory Planning. To complete the tasks of handling and assembly, we must first plan the motion of the robot, that is, action planning and trajectory planning. (1) *Action planning.* According to the design, the robot needs to carry out three subtasks: handling, assembly, and disassembly. The robot handling task refers to the robot carrying the workpieces 1–4 at the position G1 on the workpiece assembly line to the assembly area of the assembly line and recycles the pallet. The robot assembly task refers to that the robot transports the workpiece 1 of the finished product warehouse and the workpiece 2–4 of the spare parts warehouse of the assembly line to the assembly area, respectively, carries out assembly after secondary positioning, and transports the finished products to the finished product warehouse. The task of robot disassembly refers to that the robot carries the finished products in the finished product warehouse to the assembly area, disassembles them after secondary positioning, and places them at four positions in the assembly area [19]. (2) *Trajectory planning.* Trajectory planning is to decompose each action into the motion trajectory of the robot TCP. Considering the robot posture and the interference between the robot and surrounding equipment, each action needs to have one or more points to form a motion trajectory. The trajectory planning of the robot handling task is shown in Figure 7.

(2) Procedure Flow. The whole workflow of the robot handling program includes "grabbing or sucking the workpiece and pallet," "placing the workpiece and pallet," and "returning to the safety point," as shown in Figure 8, the whole workflow of the robot assembly and disassembly program.

5. System Experiment and Analysis

5.1. Experiment and Analysis of Visual Positioning Error Measurement. The accuracy of the robot visual assembly system is mainly determined by the accuracy of visual positioning and the accuracy of repeated positioning of industrial robots. The Gugao GRB-4016 industrial robot used in this system has a repeated positioning accuracy of 0.03 mm, which is high in positioning accuracy, and can meet most industrial assembly needs. For the accuracy of visual positioning, this article designs an experiment of visual positioning error measurement. In this article, a 9-point calibration block is used to measure the visual positioning error by measuring the distance between the centers of two adjacent circles and calculating the relative position between the centers.

First, use a nine-point calibration block with a height of 50 mm and place it in the field of view. Click the "multipoint measurement" button of the upper software to calculate the position of each circle center under the camera coordinate system. By calculating the difference of the coordinate positions of the adjacent two circle centers, the distance between the adjacent two circle centers can be calculated. There are 9 points on each calibration block, and 12 groups can be measured. The average value of them is the first group of data. Then, change the position of the calibration block within the field of view, measure in the upper left area, upper right area, lower left area, and lower right area within the field of view, respectively, and obtain four groups of data. Replace the nine-point calibration block with a depth of 35 mm, and the measurement method is similar to the above. The measurement errors obtained from the experiment are shown in Tables 5 and 6.

In the binocular vision measurement system, the positioning error in the x/y direction is proportional to the measurement distance and the error in the depth direction is proportional to the square of the measurement distance. Therefore, with the increase of the measurement distance, the error in the depth direction is more difficult to control than the error in the x/y direction. According to the experimental data analysis of the positioning error of the eye-to-hand binocular vision system, the error in the X direction is 0.220 mm, the error in the Y direction is 0.199 mm, and the error in the depth direction is 0.630 mm, which conforms to
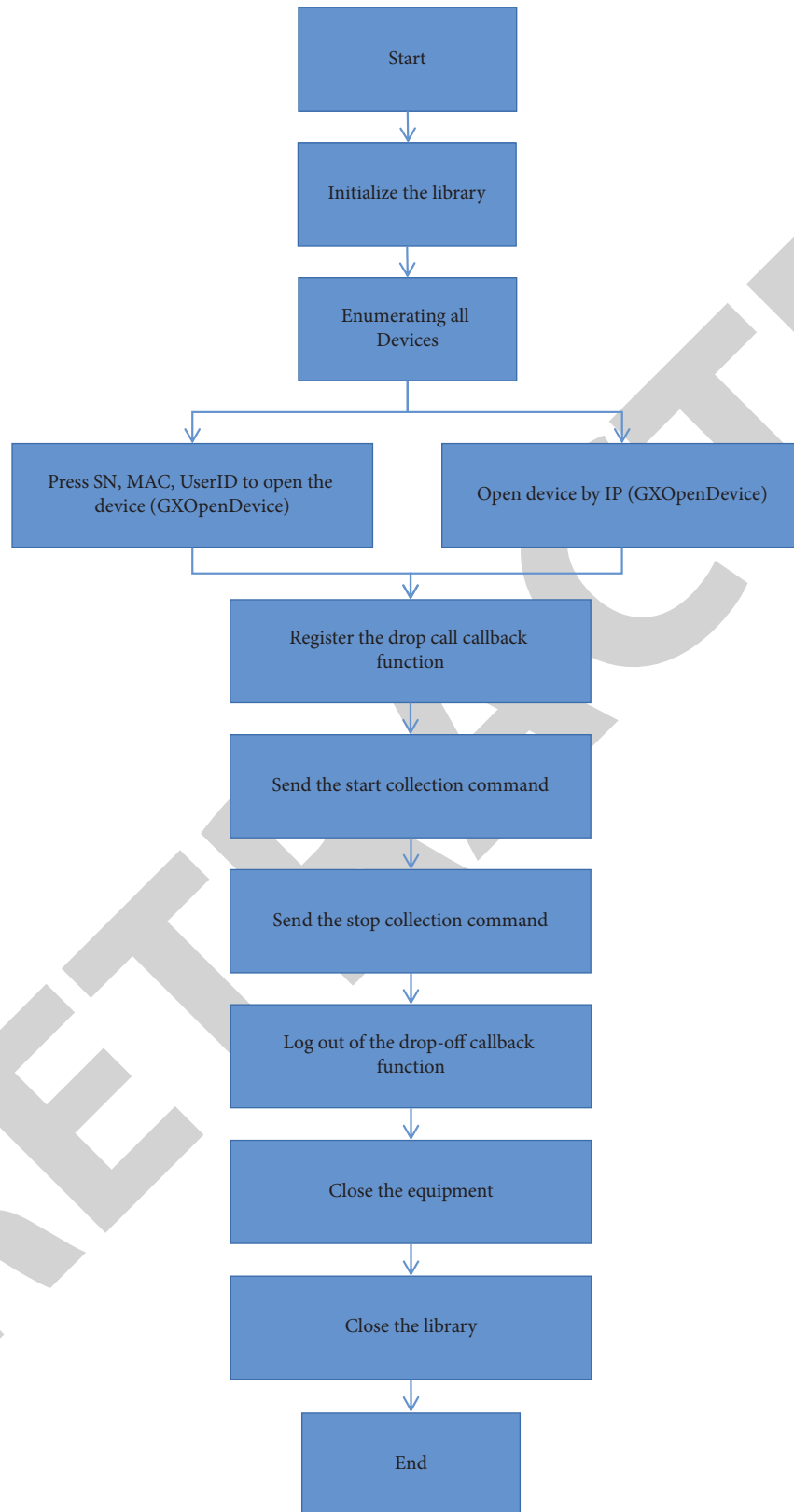
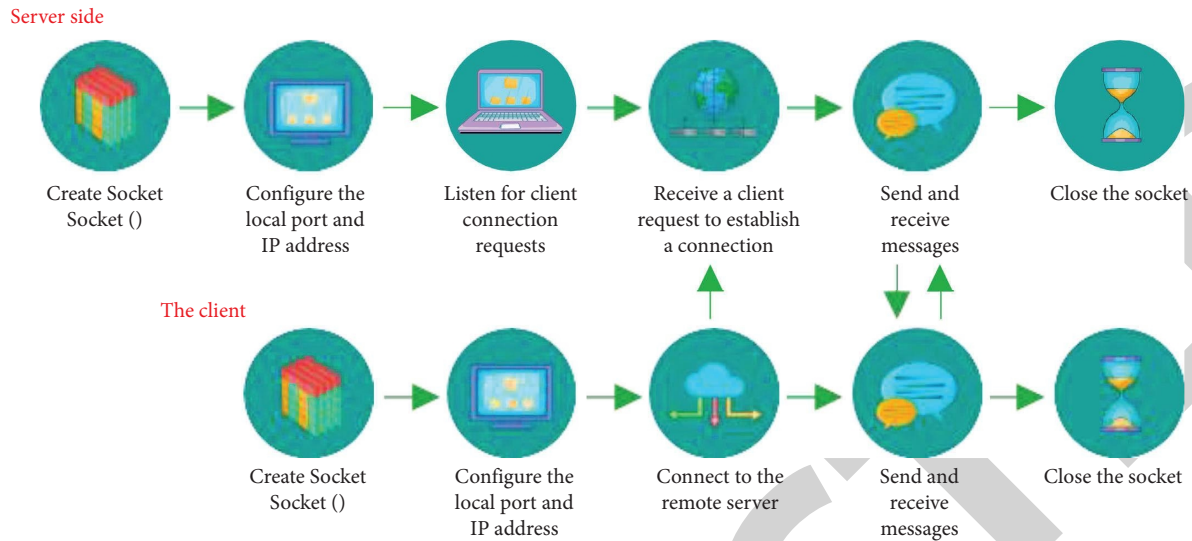Figure 4: Overall workflow of mercury camera.

Figure 5: TCP/IP communication protocol flowchart.

the characteristics of the binocular vision system measurement error. Due to the high accuracy requirements of the x/y direction of this system, in order to solve the problem that the accuracy of the eye-to-hand binocular vision system in the x/y direction cannot meet the requirements due to the long measurement distance, this article introduces the eye-in-hand monocular vision system to accurately locate the two-dimensional information of the target. According to the analysis of its experimental error data, the positioning error in the X direction is 0.033 mm and the positioning error in the Y direction is 0.045 mm.

Based on the above analysis, by introducing the eye-in-hand monocular vision system, the visual positioning accuracy in the x/y direction is greatly improved, which verifies the feasibility of the visual-positioning method based on the combination of the eye-to-hand and eye-in-hand system.

5.2. Experiment and Analysis of Overall Accuracy Measurement of the System

5.2.1. Simulation after Trajectory Optimization. On the premise that the maximum speed of the robot is constant, a five-order *s*-curve optimization method with time and impact as optimization indexes is proposed. The variable parameters are introduced into the optimization of impact force, which greatly shortens the acceleration time of the robot while ensuring the minimum impact force. In order to verify the effectiveness of the optimized fifth-order *s*-curve method, the traditional fifth-order *s*-curve and the optimized fifth-order *s*-curve are compared and analyzed, and the two curves are simulated and compared by MATLAB simulation software, so as to verify the effectiveness of the optimized curve. The simulation curves of displacement, velocity, and acceleration of the two curves are shown in Figures 9(a)–9(c).

As can be seen from the curves in Figure 9, compared with the traditional five-order *s*-curve, the acceleration time of the optimized five-order *s*-curve for robot trajectory planning is reduced from 0.65 s to 0.54 s and the maximum acceleration is

increased from 3.5 m/s^2 to 4.7 m/s^2. The acceleration time is shortened, and the motion efficiency of the robot is also improved. Therefore, the acceleration curve of the robot is smoother, greatly improving the stability of operation.

5.2.2. Overall Accuracy Measurement Results. In order to verify the overall accuracy of the robot visual assembly system, this article takes the shaft hole assembly experiment as an example. As shown in Figure 10, the axial calibration workpiece and hole calibration workpiece used in the experiment are shown. The radius of the hole calibration workpiece is 0.1 mm larger than that of the axial calibration workpiece. Therefore, as long as the positioning error is greater than 0.1 mm, the assembly cannot be successful.

First, start the six degree of freedom industrial robot, open the robot control cabinet, and check whether the servo driver of each axis starts normally. If it is normal, control the rotation of each axis through the robot teaching pendant, check whether there is any abnormality in the rotation of each axis, start the upper software of the robot visual assembly system, set the port number, wait for the robot controller to establish a connection with it, open the left and right cameras fixed on the bracket and the cameras at the end of the robot arm, and check whether the three cameras can collect images normally. After the connection is established, place the workpiece randomly on the assembly console and ensure that it is within the field of vision. The upper software controls the camera to collect the image of the workpiece and obtains the position information through the visual positioning algorithm, which is sent to the industrial robot for assembly after coordinate conversion. First, after rough positioning by the eye-to-hand binocular vision system, the robot controls the camera at the end of the manipulator to move to 15 cm above the target workpiece and triggers the camera at the end of the manipulator to take photos for secondary positioning. After secondary positioning, adjust the position of the end of the manipulator, and finally
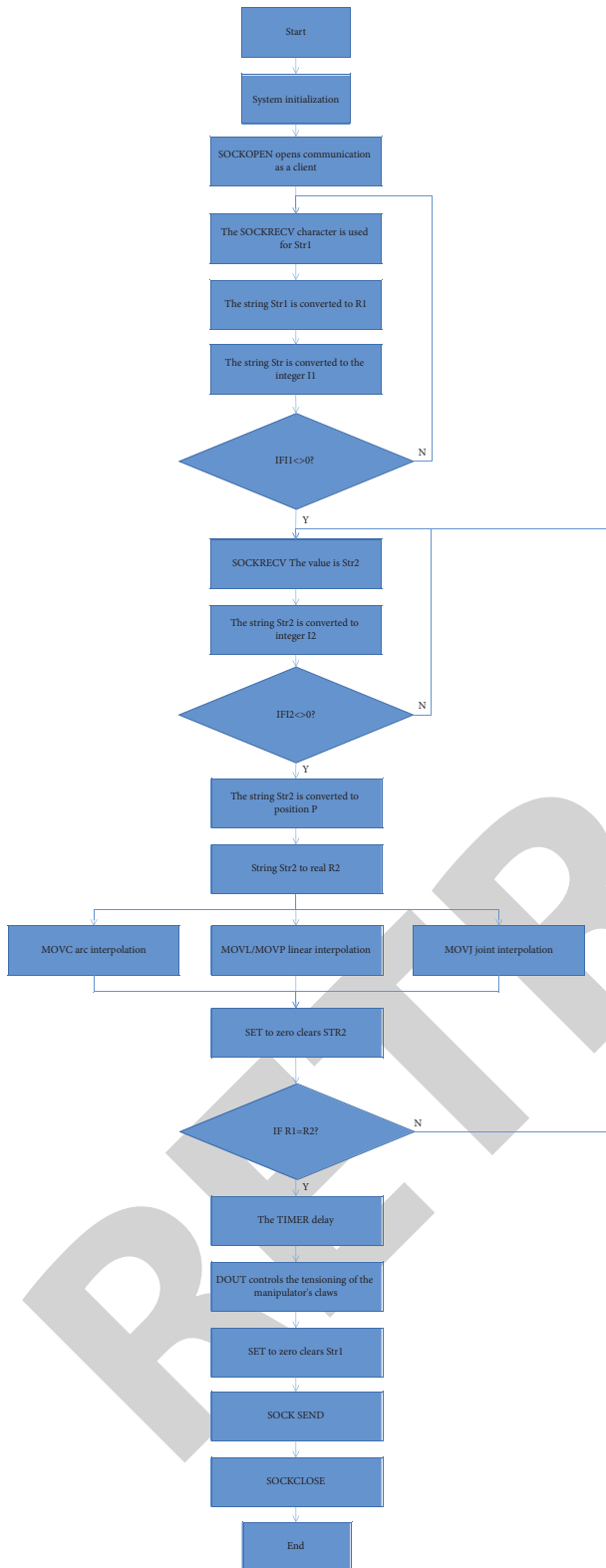
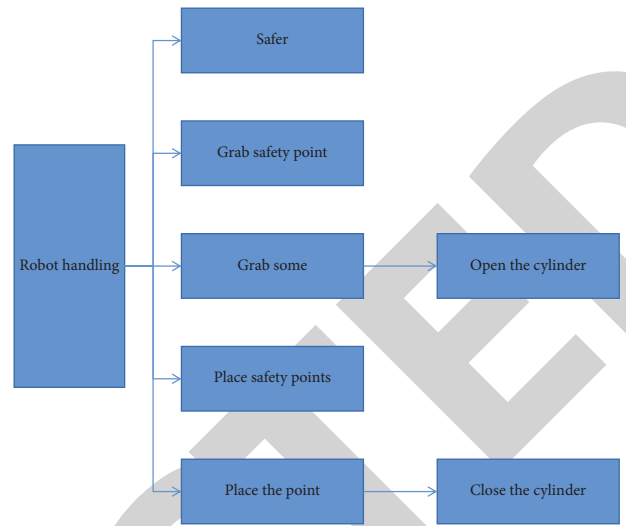Figure 6: Software design flow of the manipulator motion control module.

Figure 7: Trajectory planning.

control the assembly system for assembly operation. After assembly, the robot returns to the waiting area.

Through the analysis of the results of 10 visual guided assembly experiments at different positions and heights, the overall accuracy of the system is less than 0.1 mm in the X and Y directions and less than 1 mm in the depth direction. Through the experimental verification of the obstacle avoidance function in the assembly process of the manipulator and according to the analysis of the experimental results, it is verified that the assembly system developed in this study can autonomously avoid the obstacles encountered in the movement process when performing tasks, and the motion trajectory of the end of the manipulator is relatively smooth, there is no significant angle change, ensuring the safety in the assembly process of the manipulator [20].

This chapter mainly focuses on the experimental verification and analysis of the system function. Aiming at the accuracy measurement of the assembly system, this article first carried out the experiment of visual-positioning error measurement and measured the positioning error of the eye-to-hand binocular vision system and eye-in-hand monocular vision system through the experiment. Then, through the overall accuracy measurement experiment of the system, it is verified that the system meets the accuracy requirements of less than 0.1 mm in the x/y direction and less than 1 mm in the depth direction. Finally, the obstacle avoidance function of the manipulator in the assembly process is verified by experiments. The results show that the manipulator can successfully avoid obstacles with regular and irregular shapes in the assembly process. Through the experimental verification and analysis of the robot visual assembly system, it shows that the assembly system developed in this article has the characteristics of intelligence and high precision.
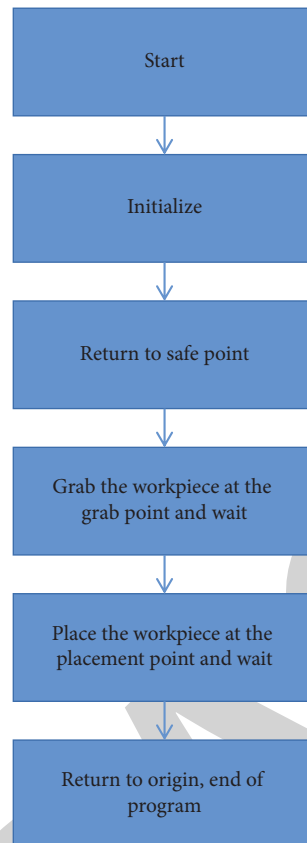
Figure 8: Procedure flowchart.

Table 5: Experimental data of the eye-to-hand binocular vision system positioning error.

Number	1	2	3	4	5	6	7	8	Average values
$\triangle x$ (mm)	0.202	0.192	0.352	0.215	0.162	0.265	0.135	0.238	0.220
$\triangle y$ (mm)	0.315	0.163	0.138	0.136	0.222	0.217	0.230	0.170	0.199
$\triangle z$ (mm)	0.713	0.825	0.695	0.912	0.835	0.652	0.325	0.625	0.630

Table 6: Experimental data of the eye-in-hand monocular vision system positioning error.

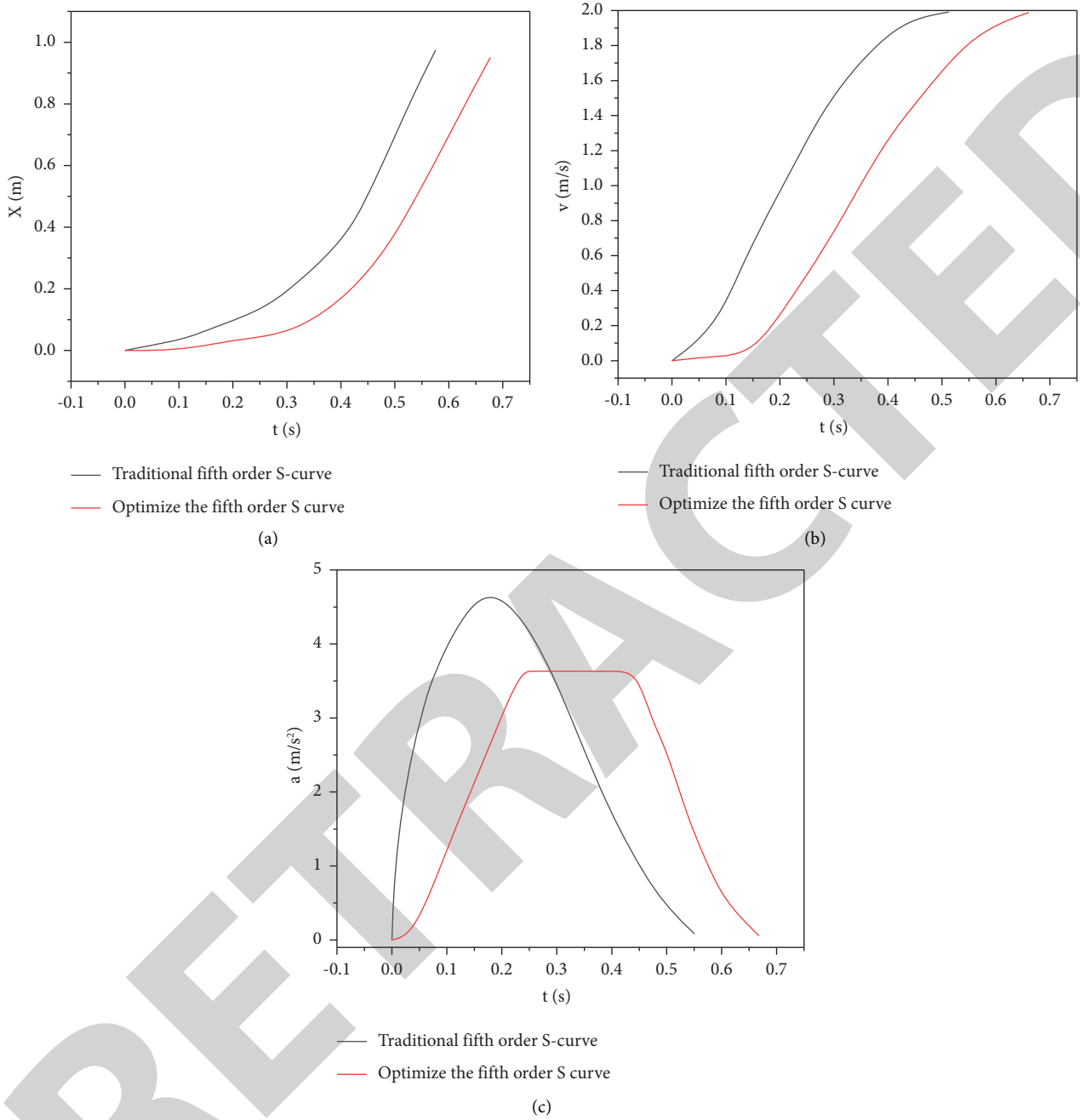| Number	1	2	3	4	5	6	7	8	Average values
$\triangle x$ (mm)	0.017	0.040	0.022	0.072	0.069	0.017	0.014	0.026	0.033
$\triangle y$ (mm)	0.034	0.008	0.050	0.070	0.081	0.009	0.009	0.092	0.045

(a)

(b)

(c)

Figure 9: Simulation curves of different algorithms: (a) displacement curve, (b) velocity curve, and (c) acceleration curve.
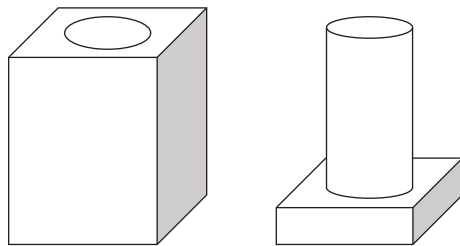

Figure 10: Experimental workpiece.

6. Conclusion

This article takes the industrial robot assembly technology based on machine vision as the research object, mainly studies the visual measurement technology and the robot arm obstacle avoidance technology in the assembly process, and shows the theoretical research results by developing the robot visual assembly system. The system can be flexibly applied in complex and changeable working environment, which greatly improves the practical application range of the industrial robot assembly technology.

(1) According to the analysis of the system function and performance index requirements, the visual-positioning method combining the eye-to-hand and eye-in-hand system is used to measure the position of the target. By studying camera calibration technology, robot hand eye calibration technology, and image feature extraction technology, the accurate positioning of the target workpiece is achieved.

(2) The obstacle avoidance problem of the manipulator in the assembly process is studied. Firstly, the obstacle circle is used.

The cylindrical minimum bounding box models various irregular obstacles and realizes the visual location of obstacles through the minimum bounding circle algorithm. Combined with the obtained obstacle information and target workpiece information, the artificial potential field method is used to realize the obstacle avoidance path planning of the manipulator.

(3) The feasibility of the visual-positioning method used in this system is verified by setting up and completing the visual positioning error measurement experiment and the overall accuracy measurement experiment of the system. The obstacle avoidance function of the system is verified by the obstacle avoidance function verification experiment of the manipulator.

The system adopts the visual-positioning method combining the eye-to-hand and eye-in-hand system to improve the positioning accuracy in the X and Y directions of the system, and the depth information is measured by eye-to-hand binocular vision system. Through the experimental verification of the positioning error based on the eye-to-hand binocular vision system and eye-in-hand monocular vision system, the system proposed in this article meets the design accuracy requirements of less than 0.1 mm in the x/y direction and less than 1 mm in the depth direction. Considering that the currently developed binocular vision system is not accurate enough in measuring the target depth information, the next step will be to study the methods to improve the measurement accuracy of the depth direction.

Data Availability

The labeled data set used to support the findings of this study is available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest.

Acknowledgments

References

[1] J. Zhou and J. Yu, "Chisel edge wear measurement of high-speed steel twist drills based on machine vision," *Computers in Industry*, vol. 128, no. 5–8, Article ID 103436, 2021.

[2] B. Lza, C. Jian, A. Zy, Y. A. Hao, A. Mc, and B. Xy, "Research on precision automatic tool setting technology for kdp crystal surface damage mitigation based on machine vision," *Journal of Manufacturing Processes*, vol. 64, no. 2, pp. 750–757, 2021.

[3] X. Lin, X. Wang, and L. Li, "Intelligent detection of edge inconsistency for mechanical workpiece by machine vision with deep learning and variable geometry model," *Applied Intelligence*, vol. 50, no. 7, pp. 2105–2119, 2020.

[4] M. Talaat, I. Arafa, and H. Metwally, "Advanced automation system for charging electric vehicles based on machine vision and finite element method," *IET Electric Power Applications*, vol. 14, no. 13, pp. 2616–2623, 2020.

[5] R. Kumar, O. Patil, K. Nath S, K. S. Sangwan, and R. Kumar, "A machine vision-based cyber-physical production system for energy efficiency and enhanced teaching-learning using a learning factory," *Procedia CIRP*, vol. 98, no. 9, pp. 424–429, 2021.

[6] Y. Zhang, H. G. Soon, D. Ye, J. Y. H. Fuh, and K. Zhu, "Powder-bed fusion process monitoring by machine vision with hybrid convolutional neural networks," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 5769–5779, 2020.

[7] Y. Guo, L. Chai, S. E. Aggrey, A. Oladeinde, J. Johnson, and G. Zock, "A machine vision-based method for monitoring broiler chicken floor distribution," *Sensors*, vol. 20, no. 11, pp. 3179–3183, 2020.

[8] Z. Liu, Y. He, C. Wang, and R. Song, "Analysis of the influence of foggy weather environment on the detection effect of machine vision obstacles," *Sensors*, vol. 20, no. 2, pp. 349–352, 2020.

[9] C. Li, Y. Huang, H. Li, and X. Zhang, "A weak supervision machine vision detection method based on artificial defect simulation," *Knowledge-Based Systems*, vol. 208, no. 4, Article ID 106466, 2020.

[10] T. Zhang, X. Hu, J. Xiao, and G. Zhang, "A machine learning method for vision-based unmanned aerial vehicle systems to understand unknown environments," *Sensors*, vol. 20, no. 11, pp. 3245–3250, 2020.

[11] A. M. Karimi, J. S. Fada, N. A. Parrilla et al., "Generalized and mechanistic pv module performance prediction from computer vision and machine learning on electroluminescence images," *IEEE Journal of Photovoltaics*, vol. 10, no. 3, pp. 878–887, 2020.

[12] Z. Jiang, M. Huang, X. Tang, and Y. Guo, "A new calibration method for joint-dependent geometric errors of industrial robot based on multiple identification spaces," *Robotics and Computer-Integrated Manufacturing*, vol. 71, no. 1, pp. 102175–102182, 2021.

[13] F. Mohammadi Amin, M. Rezayati, H. W. van de Venn, and H. Karimpour, "A mixed-perception approach for safe human–robot collaboration in industrial automation," *Sensors*, vol. 20, no. 21, pp. 6347–6351, 2020.

[14] T. Thalmann and H. Neuner, "Temporal calibration and synchronization of robotic total stations for kinematic multi-sensor-systems," *Journal of Applied Geodesy*, vol. 15, no. 1, pp. 13–30, 2021.

[15] X. Ren, C. Li, X. Ma et al., "Design of multi-information fusion based intelligent electrical fire detection system for green buildings," *Sustainability*, vol. 13, no. 6, p. 3405, 2021.

[16] R. Huang, "Framework for a smart adult education environment," *World Transactions on Engineering and Technology Education*, vol. 13, no. 4, pp. 637–641, 2015.

[17] J. Hu, Y. M. Kang, Y. H. Chen, X. Liu, X. Li, and Q. Liu, "Analysis of aerosol optical depth variation characteristics for 10 years in urumqi based on modis_c006," *Huan jing ke xue= Huanjing kexue*, vol. 39, no. 8, pp. 3563–3570, 2018.

[18] J. Jayakumar, B. Nagaraj, S. Chacko, and P. Ajay, "Conceptual implementation of artificial intelligent based E-mobility controller in smart city environment," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 5325116, 8 pages, 2021.

[19] G. Li, F. Liu, A. Sharma et al., "Research on the natural language recognition method based on cluster analysis using neural network," *Mathematical Problems in Engineering*, vol. 2021, Article ID 9982305, 13 pages, 2021.

[20] K. Xia, C. Saidy, M. Kirkpatrick, N. Anumbe, A. Sheth, and R. Harik, "Towards semantic integration of machine vision systems to aid manufacturing event understanding," *Sensors*, vol. 21, no. 13, p. 4276, 2021.