Journal of Computers Vol. 35 No. 4, August 2024, pp. 207-224
doi: 10.53106/199115992024083504015

Method and System Design for Industrial Robots to Grip Stacked and Disorder Parts Under 3D Visual Guidance

Qing-Chuan Liu[1,2], Jian Gao[1], Rui Fan[1],
Jun-Xia Cui[1*], and Wei-Min Liu[1]

[1] Hebei Institute of Mechanical and Electrical Technology,
Xingtai City 054000, Hebei Province, China
{qingchuan3978, gaojian87989, fanrui98792, xia333898, weimin78687}@126.com

[2] Xingtai Intelligent Production Line and Equipment Technology Innovation Center,
Xingtai City 054000, Hebei Province, China

Received 19 July 2024; Revised 31 July 2024; Accepted 15 August 2024

Abstract. With the production of bulk parts and a large number of assembly scenarios, most parts are stacked in an unordered manner. This article focuses on the accurate grasping of parts in an assembly line under the guidance of binocular vision. Firstly, based on production needs, this article designs a robot grasping system based on binocular vision, and describes the selection of key equipment such as cameras and robotic arms in the system. Then, in order to achieve the recognition of stacked parts, analysis and research were conducted on point cloud data extraction, point cloud feature recognition, point cloud registration, and part pose estimation in the recognition process. Finally, through the construction of a simulation system, the recognition of stacked parts was achieved, and the recognition accuracy was improved.

Keywords: 3D vision, industrial robot, part grabbing, pose estimation

1 Introduction

With the rapid development of intelligent manufacturing technology and robotics, many fields such as mechanical production, electronic products, logistics and transportation, chemical and medical industries are transitioning towards intelligent manufacturing. The implementation of intelligent manufacturing cannot be separated from robot technology and visual recognition technology. The application and research of machine vision guided robotic arms in automatic assembly can promote the development of China's manufacturing industry and have important practical significance for improving the quality of intelligent manufacturing in China. It meets the requirements of the "14th Five Year Plan" for the intelligent and digital transformation of equipment manufacturing industry in China. At the same time, with the changes in the international situation and the adjustment of China's strategy, the intelligent manufacturing supply chain in China and even the Asia Pacific region can achieve complementary advantages and mutual promotion through restructuring. While improving economic development in the context of economic downturn, it also provides many opportunities for cost reduction and efficiency improvement of China's manufacturing industry and the development of production scale. Currently, China's robot research and machine vision technology are still in progress. In the early stages of development. There are still many issues that need further research in the application of machine vision [1].

The use of industrial robots for automatic assembly has broad application prospects, and in recent years, researchers have continuously deepened their research on automatic control, path planning, and improvement of repetitive positioning accuracy of industrial robots. Early industrial robots' semi-automatic assembly was achieved through offline programming or teaching methods. Each time the robot grabbed a part from a fixed and identical position, the pose of the target part was required to be consistent. Once there was a deviation in the position of the part, subsequent actions could not be completed, making it difficult to meet the production requirements of current flexible assembly and production line universality. Therefore, achieving adaptive acquisition of part types and poses by robots, realizing active grasping, and developing robot automatic assembly are of great significance for industrial upgrading and intelligent development of enterprises [2].

Stereoscopic vision technology is a technical concept proposed by Roberts in the 1960s [3]. The technical

* Corresponding Author

principle is to obtain the three-dimensional information of an object in a two-dimensional image, and then construct the depth and size information of the object to achieve deeper positioning of parts. Currently, in terms of stereoscopic vision applications, stereoscopic vision technology can be divided into two types: active vision and passive vision. The principle of stereoscopic vision is shown in Fig. 1.

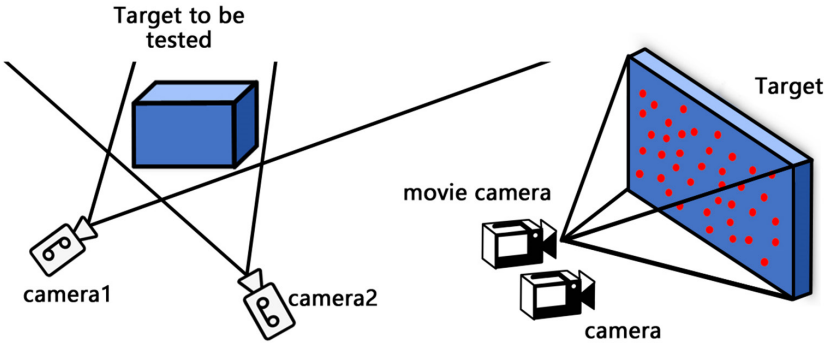

Fig. 1. Principles of 3D vision

Stereoscopic vision technology is an important research direction in the field of machine vision. It uses stereo vision sensors to obtain three-dimensional information of target objects in the scene. Based on its advantages of high accuracy and fast speed, stereo vision, as the "eye" of industrial robots, has been widely used in the field of intelligent robot assembly and handling.

Therefore, this article focuses on the precise grasping of stacked parts by industrial robots guided by stereo vision. The work done is as follows:

1) Based on the analysis of actual production scenarios, the overall design of the 3D vision grasping system was completed, and detailed descriptions were given for the selection of binocular vision cameras, industrial robots, and robot grippers in the system. At the same time, the overall structure of the software was designed, and the parameters of each functional module were clarified according to production needs.

2) Before grasping, the robot needs to accurately identify stacked parts. For the recognition of stacked parts, it mainly completes the acquisition of part point clouds, feature recognition of part point clouds, registration of part point clouds, and pose estimation. Finally, an improved neural network is used to improve the pose estimation efficiency of the parts.

3) Finally, a simulation environment was set up to validate the method proposed in this paper.

2 Related Work

There are relatively few research results on the grasping of stacked parts by industrial robots guided by 3D vision, but scholars have done a lot of work on the grasping and assembly of non stacked arbitrary pose parts, providing many technical methods for this article.

Hua Luo from Northwestern Polytechnical University used the surface structured light method to construct Sangwei point cloud data for large-scale equipment. In the process of part recognition, point cloud registration technology was used to identify the structural features of the parts and complete pose estimation. An improved hand eye calibration strategy was used to guide the robot to grasp parts in any pose. Finally, the entire grasping system was verified in simulation environment and real experimental scene. The translation error of the entire grasping system can be reduced to 0.413mm, and the angle error can be reduced to 0.123 °. The proposed method can quickly and effectively recognize and locate scattered stacked parts with high precision, guiding industrial robots to accurately grasp and place them [4].

Ping Chen from Chongqing University mainly focuses on grasping parts containing circular holes. In the process of 3D vision estimation of the pose of circular hole parts, a three-dimensional point cloud based axis pose estimation algorithm is proposed. Firstly, based on the key point selection method of the three-dimensional point

cloud, the algorithm for rough axis estimation is proposed and analyzed based on the geometric constraints of the surface normal of the point cloud and the axis. Then, in order to achieve the estimation of the pose of the part, an iterative robust least squares based axis pose optimization algorithm is used to estimate the pose of the circular hole part. Finally, after experiments, the root mean square error of the axis pose estimation angle is 0.248 ◦, and the root mean square error of the position is 0.463 mm, which is compared with the existing popular axis estimation methods. The proposed method has higher accuracy. The assembly strategy can meet the production requirements of high precision, stability, and reliability in the assembly of robot circular axis hole parts [5].

Youfa Xu from China Electronics Technology Group obtained a three-dimensional visual matrix based on the random sampling consistency rule (RANSAC) and the near point iteration method (ICP) template matching algorithm. The robot grasping method is guided by the robot grasping point and the end robot grasping pose. At the same time, the method designs a pre grasping position for the robot. The robot first adjusts the grasping posture to reach the pre grasping position during grasping, and then runs to the grasping point to grasp the object while keeping the grasping posture unchanged. Finally, an experimental scene is designed. The experimental results show that the method of combining the matrix obtained by the template matching algorithm with the actual grasping of the robot can grasp any placed object through the matching matrix, making the theoretical research of point cloud landing in practical grasping applications, and improving the grasping accuracy. Improved, The proposed method meets production needs [6].

Xudong Cui uses machine vision for two-dimensional and three-dimensional measurement and information fusion, and employs binocular stereo vision technology to measure and obtain three-dimensional point cloud data of workpieces in a scattered stacking state; By segmenting the entire 3D point cloud data, obtain the 3D data belonging to the visible part of a single workpiece under the condition of attention concentration; Finally, by discretizing the three-dimensional data of a single workpiece with the established CAD model, the three-dimensional point cloud matching is performed to determine the three-dimensional pose of the target workpiece in the scattered stacking, and automatically generate the motion trajectory and action program for controlling the six joint robot to perform grasping, thus completing the sorting work of the workpiece [7].

Jie Tao established a 3D machine vision based disorderly grasping system for stacked components based on the existing water pump production line. The global feature descriptor PPF algorithm was used to extract component features, and pairwise combinations of visible points that met the conditions were performed. The RANSAC algorithm was then applied for rough pose estimation, and the ICP algorithm was used to fine tune the pose to obtain the optimal pose estimation of the target. The position and posture of the components in the real-world coordinate system were then determined to guide the robotic arm to accurately grasp and place the stacked components. The experimental results showed that the average success rate of overall grasping could reach 92.3% [8].

Shengyin Zhu from Geely Automobile Group proposed an irregular part grasping method based on 3D vision robotic arm to solve the problem of inaccurate positioning or low grasping accuracy of parts. The method mainly includes system space calibration, part point cloud positioning and segmentation, and part pose estimation algorithm modules. He also proposed a point cloud camera extrinsic calibration method based on 2D images and a part point cloud segmentation method based on image semantic segmentation. Through experimental verification, the proposed calibration method achieves a system accuracy error of 1 mm, and has high solving efficiency and strong universality; Compared with commonly used clustering segmentation, the point cloud segmentation method for parts has higher accuracy and efficiency. The two-stage part spatial pose calculation method designed can accurately estimate the 6-degree-of-freedom pose of the part in space, which is about 1.8 times more accurate than the single-stage matching method. The proposed 3D vision based robotic arm irregular part grasping system has high accuracy, strong scalability, and wide adaptability [9].

In summary, scholars have achieved good research results on the grasping work of 3D vision guided robots for arbitrary poses, stacked and non stacked parts, providing technical support for the establishment of grasping strategies in this paper. However, in the process of analyzing the methods and strategies of various scholars and combining them with existing production practices, the accuracy and speed of point cloud feature recognition for stacked parts by industrial robots under 3D vision guidance still need to be improved.

Therefore, this article focuses on the recognition method of stacked parts and the design of the overall grasping system. In order to describe the work done in this article clearly, it is divided into the following chapters: Chapter 2 mainly introduces the relevant research results, providing technical support for the improvement of the method strategy in this article; Chapter 3 is the construction of the 3D vision system, focusing on the design of core links such as cameras and lenses, as well as the selection of hardware equipment; Chapter 4 mainly describes the recognition process of parts, including point cloud feature extraction, feature recognition, feature matching and other processes; Chapter 5 constructs a simulation experimental environment and verifies the method in this article. Through comparison, the performance improvement of this method is demonstrated; Chapter

6 is the conclusion part. Summarizing the work done in this article, and prospects for future research directions were also discussed.

3 Design of 3D Visual Robot Grasping System

This article takes the assembly production line of new energy vehicle parts as the working scenario, and focuses on the grabbing of shaft and gear parts in the production line as the research object. In traditional automated production lines, industrial robot controllers mostly control industrial robots to complete the assembly process of structural parts such as gears and shafts through offline programming or teaching reproduction methods, requiring the grabbing target to be a relatively fixed position and fixed pose. The flexibility of the entire recognition grabbing process is poor, and the flexibility of the industrial robot grabbing system is relatively low. This article addresses the above issues and designs an industrial robot autonomous grasping system that can be widely applied in various unstructured environments and disorderly stacking scenarios. The overall layout of the system is shown in Fig. 2.

1) The visual inspection system consists of a computer as the control core and an IDS uEye CP camera connected via USB 3.0 interface, forming the visual inspection system of the robotic arm. The visual inspection system mainly uses binocular cameras to recognize object features;

2) The communication system mainly completes the communication task between the computer and the robotic arm. The computer is connected to the industrial robot control cabinet through Ethernet, establishing TCP/IP communication between the two. The industrial robot control cabinet controls the opening and closing of the end electric gripper through I/O signals.

3) The grasping system consists of industrial robots and electric or pneumatic grippers, serving as the executing part of the entire automatic grasping process. Upon receiving control signals, it automatically adjusts the grasping pose to grasp the target part.

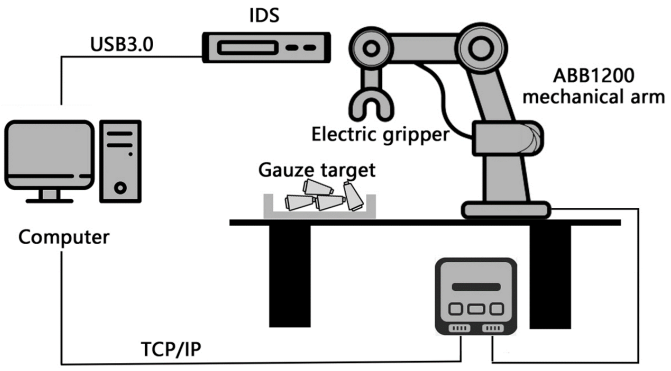

Fig. 2. Composition of 3D visual grasping system

The entire system needs to ensure the accuracy and speed of recognition during the working process. Firstly, the depth camera of the system is calibrated to obtain the internal parameters of the camera. Then, hand eye calibration is performed between the camera and the industrial robot to obtain the coordinate conversion relationship between the camera and the industrial robot.

The 3D camera captures depth images of scenes containing parts such as gears and shafts, converts the depth images into 3D point cloud data, and performs a series of processing such as filtering, segmentation, feature extraction, and registration to obtain the final position and orientation information of stacked parts. After that, the six degree of freedom industrial robot is guided to complete the grasping task of shaft and gear parts [10].

This article uses commonly used shafts in automated production lines as the target to be grasped by the robotic arm. A 3D camera is used as the data input source to obtain three-dimensional information of the shaft parts, in-

cluding depth information and position information. The detection and positioning of gear targets are completed through computer vision and image processing technology. The robotic arm then grasps the target based on the target pose information obtained by the vision system [11].

This chapter mainly describes the selection of devices in the system that determine the accuracy of grasping tasks. According to the design plan, the hardware equipment required for this article mainly includes IDS uEye CP depth camera, ABB1200 robotic arm, self-designed electric gripper and computer (PC), as well as other communication auxiliary equipment.

3.1 Selection of Depth Cameras

This article uses the second-generation IDS uEye CP camera, which is a powerful and compact stereoscopic 3D camera. The camera operates based on the process of "projection texture stereo vision" and has two integrated CMOS sensors. The camera is made of aluminum shell, with a size of 29×29×29 millimeters. The working voltage of the camera is 12-24 V, and the camera parameters are shown in Table 1.

Table 1. IDS uEye CP camera parameter list

Parameter	Parameter values
Pixel	2.35MP
Resolution ratio	1936×1216
Frame rate	41fps
Size of photosensitive surface	11.34×7.13*mm*
Pixel size	5.86μm

The infrared emitter of IDS uEye CP camera continuously projects infrared light into objects within the field of view. At this time, the sensor's internal timer calculates the emission time of each group of beams. When the infrared detector searches for signals reflected back from the surface of the object being measured, the timer stops counting and measures the depth information of each pixel on the surface of the object by calculating the time difference between the emission and return beams. The calculation formula is as follows:

$$d = \frac{c}{2} \cdot \frac{\Delta\varphi}{2\pi f}. \tag{1}$$

In the formula, d represents the distance from the target point to the camera plane, $\Delta\varphi$ represents the phase difference between the same incident and reflected light, c is the known propagation speed of light, and f is the frequency of the sensor.

In practical applications, the depth deviation and disparity deviation of binocular vision objectively exist. Assuming the depth deviation is Δd and the visual deviation is Δs, the following formula can be obtained:

$$d + \Delta d = \frac{B \cdot f}{S + \Delta s}. \tag{2}$$

In the formula, B is the baseline distance, which is the distance between the optical centers of the left and right cameras.

From the above formula, it can be concluded that the smaller the pixel size of the camera sensor, the higher the depth accuracy. If B, f, and Δd remain constant, then S becomes smaller, indicating that the closer the measurement point is to the binocular vision system, the higher the depth accuracy. Therefore, in the process of camera installation design, ideal accuracy should be achieved through reasonable spacing settings [12].

3.2 Mechanical Gripper Selection

The parameters of ABB industrial robots in the entire visual grasping system are very detailed, so this article will not go into too much detail about the parameters of ABB robots. The installation of robot electric grippers and

the end flange of industrial robots are non-standard tools that require structural design based on actual work requirements.

In the process of designing the gripper, two main parameters are considered, namely the stroke and gripping force of the gripper. Therefore, in this article, the gripper drive motor is driven by a stepper motor with an encoder to ensure the size of the gripping force while ensuring stroke accuracy control. When the gripper is in working condition, automatic reset can be achieved through the controller, and the gripper can automatically adjust the gripping force in real time based on the gripping force model. It can also detect and correct the travel boundary through its own carried sensors. At the same time, the gripper also has data transmission function, which can transmit signals such as the working status, size measurement results, and abnormal working alarms of the gripper to the control system through the communication system. Pre set the movement stroke, direction, and speed of the gripper through USB and I/O communication between the electric gripper controller and the robotic arm control cabinet. The parameters of the gripper are shown in Table 2.

Table 2. Specification parameters of electric gripper

Parameter	Parameter values
Two claw stroke	35*mm*
Clamping force	70-100*N*
Speed	1-50*mm/s*
Repeatability	0.01*mm*
Weight	0.67*kg*
Source of driving force	Stepping motor

3.3 Design of Grab System Software Platform

The software system is the prerequisite for the robotic arm to achieve its predetermined functions and the carrier of visualization functions. By connecting the above hardware systems through control programs, the system can operate in an orderly manner. This article takes the Windows 11 operating system as the background, uses Matlab simulation software for program design, and adopts a modular approach to build a 3D vision based robotic arm grasping software system. According to the functional requirements of each module, the software system is divided into interactive operation layer, data processing layer, physical interface layer, and basic dependency layer [13]. The schematic diagram of the structure is shown in Fig. 3.

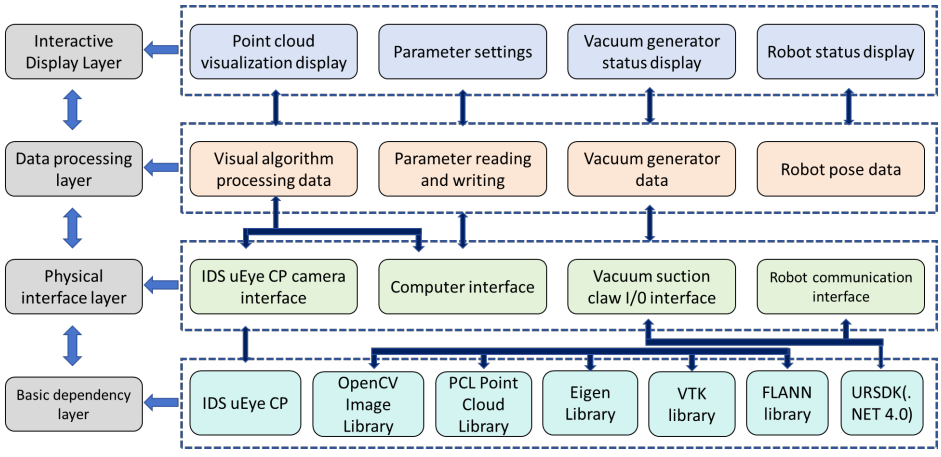

Fig. 3. Software system composition

The various functional modules in the system coordinate with each other and transmit signals through corresponding communication protocols. The following are the roles of each layer:

1) Interactive display layer: mainly realizes the setting of global system parameters, display of robot status parameters, visual display of point cloud data during operation, and printing of important logs.

2) Data processing layer: It is the part where the computer obtains device data and further processes the data, mainly including the parameter reading and writing module, camera data processing, vacuum generator state data processing, robot pose data processing, and visual algorithm data processing.

3) Physical interface layer: mainly includes IDS uEye CP camera interface, computer interface, vacuum gripper I/O interface, and robot communication interface.

4) Basic dependency layer: mainly includes camera SDK, robot SDK, OpenCV image library, PCL point cloud library, Eigen library, VTK library, FLANN library and other dependency libraries, which facilitate the rapid deployment of algorithm modules and data processing.

After the above process, this article has completed the design of the hardware and software structure of the 3D robot grasping system. At the hardware level, it mainly consists of three systems: visual inspection system, communication system, and grasping system. Then, a visual inspection system framework was built, which serves as the control framework of the entire grasping system and can display the system status, recognize grasping results and other information to help users complete grasping control.

4 Feature Recognition and Pose Estimation of Parts

The recognition of stacked parts follows the following steps: point cloud feature extraction, point cloud feature point preprocessing, point cloud keypoint extraction, and stacked part registration, which includes point cloud segmentation and pose estimation. Point cloud feature extraction is the foundation of recognition, and pose estimation is the key to determining the accuracy of final part recognition. Therefore, in the pose estimation stage, intelligent algorithms are introduced to provide registration accuracy for part poses.

4.1 Point Cloud Feature Extraction

In order to improve computational efficiency, it is necessary to preprocess point cloud data in practical point cloud applications. 3D point cloud keypoint extraction involves obtaining stable and iconic feature points from the target point cloud data through certain detection algorithms, serving high-level visual algorithms such as object tracking, registration, modeling, spatial structure description, and object recognition.

Excessive number of point clouds can bring difficulty to the segmentation processing of the target point cloud. To increase the efficiency of subsequent algorithms and further address the problem of large amounts of original point cloud data, this paper effectively reduces the number of point clouds through voxelization filtering [14].

1) Based on the coordinate information of the original point cloud data, calculate the maximum and minimum values on the three coordinate axes x, y, and z.

2) Set the edge length r of the voxel grid;

3) Calculate the side lengths h_x, h_y, h_z of the minimum bounding box of the point cloud based on the maximum and minimum values on the three coordinate axes. The expression is as follows:

$$\begin{cases} h_x = x_{\max} - x_{\min} \\ h_y = y_{\max} - y_{\min} \\ h_z = z_{\max} - z_{\min} \end{cases} \tag{3}$$

4) In the formula, $(x, y, z)_{\max}$ is the maximum value of the coordinate axis, and $(x, y, z)_{\min}$ is the minimum value of the coordinate axis.

The voxel grid size is represented as:

$$\begin{cases} L_x = \left| h_x - r \right| \downarrow \\ L_y = \left| h_y - r \right| \downarrow \\ L_z = \left| h_z - r \right| \downarrow \end{cases} \tag{4}$$

Sort all elements from small to large, calculate the centroid of each voxel grid, and replace all points in the entire voxel grid with the centroid. After processing, the sampling results of the shaft parts are shown in Fig. 4.

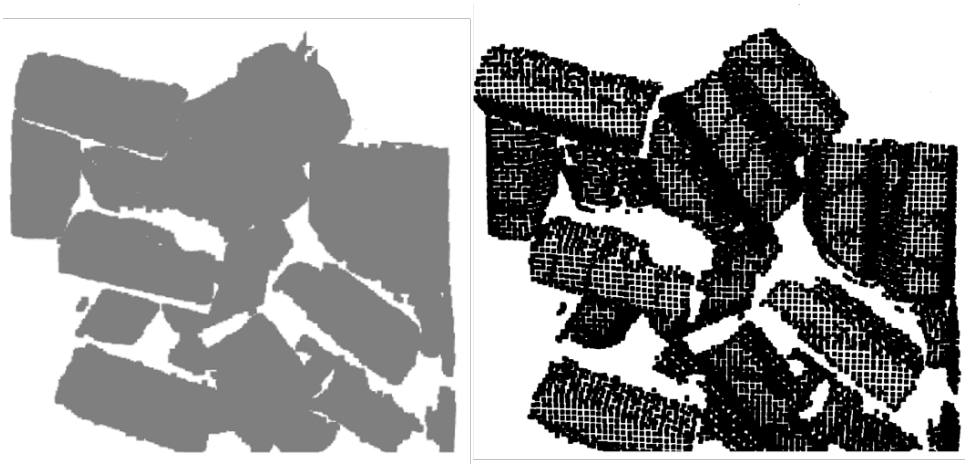

Fig. 4. Image processing results

Due to the uneven density of the point cloud and the presence of sparse outliers in the scene point cloud due to the errors of the collection equipment, considering the characteristics of outliers, this paper adopts a statistical filtering method to filter them out. *k–d* tree is used to query the scene point cloud, and the distance between spatial points and neighboring points is calculated. If the distance between each point is Gaussian distribution, the outlier value of the distance can be calculated by the mean μ and standard deviation σ, and the outlier value can be filtered out [15]. The steps to remove outliers are as follows:

1) Set a point $A_n(x_n, y_n, z_n)$ in the point cloud, and the distance from that point to any point $B_m(x_m, y_m, z_m)$ is expressed as:

$$S_t = \sqrt{(x_n - x_m)^2 + (y_n - y_m)^2 + (z_n - z_m)^2}. \tag{5}$$

2) Traverse all points and calculate the mean A and standard deviation B of the distance between that point and any point:

$$\mu = \frac{1}{n}\sum_{t=1}^{n} S_t. \tag{6}$$

$$\sigma = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(S_t - \mu)^2}. \tag{7}$$

3) Set thresholds λ and δ, where λ represents the number of neighboring points in a point cloud data, and δ is a multiple of the standard deviation σ. The method for preserving and proposing points is as follows:

$$\begin{cases} \mu \le (\mu - \lambda * \delta, \mu + \lambda * \delta), \text{reserve} \\ \mu > (\mu - \lambda * \delta, \mu + \lambda * \delta), \text{eliminate.} \end{cases} \tag{8}$$

The image filtered out by outliers is shown in Fig. 5.

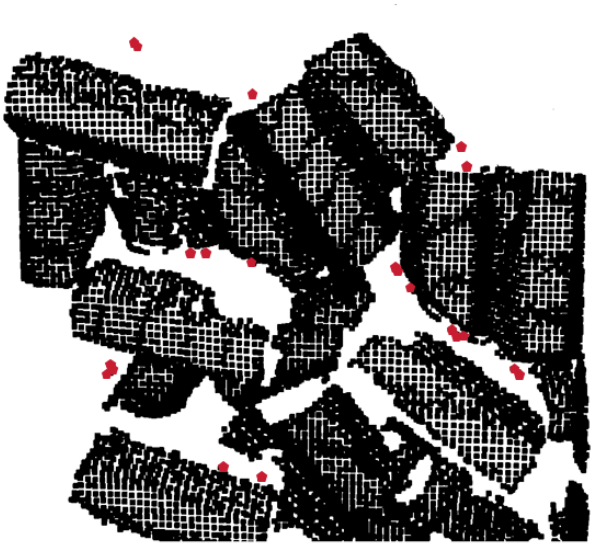

Fig. 5. Outlier filtering

4.2 Key Point Cloud Extraction

Set the point cloud dataset $P = \{p_m, m = 0 \ldots n-1\}$, where the current point is A_i. First, use the k–d-tree algorithm to obtain its k nearest neighbor points. Set the nearest neighbor point set to $N(i)$ [16], and then calculate the saliency features of the point in three steps:

1) Calculate the average composite vector of point A_i and its nearest neighbors. For each point A_j in $N(i)$, let the vector pointing from point A_j to the current point A_i be $\overrightarrow{f_{ji}}$, normalize $\overrightarrow{f_{ji}}$ to the unit vector $\overrightarrow{p_{ji}}$, and obtain the vector set $\overrightarrow{f_{ji}} = \left[f_{ji}, A_j \in N(i), j \neq i \right]$. Let the average composite vector of k nearest neighbors of the current point A_i be f_i, then:

$$f_i = \frac{1}{t} \sum_{\substack{j=1 \\ j \neq i}}^{t} p_{ji}. \tag{9}$$

2) Calculate the local taper feature of point A_i, normalize $\overrightarrow{f_i}$ to the unit vector $\overrightarrow{p_i}$, calculate the inner product of each vector and $\overrightarrow{p_i}$ in the vector set, obtain the inner product set $I = \left[c_{ji}, A_j \in N(i), j \neq i \right]$, and select the minimum value c_i from it.

$$c_i = \min_{j \in N(i)} \{c_{ji}\}, c_i \in [-1,1]. \tag{10}$$

c_i is the cosine value of the maximum angle between the difference vector of point A_i and its neighboring points, which deviates from its mean composite vector. Its magnitude approximately reflects the size of the taper of the cone, while its positive and negative values reflect the outward convex or inward concave characteristics of the cone. When $c_i > 0$ represents the area near point A_i with outer cone features, and if $c_i < 0$ represents the area near point A_i with inner cone features.

3) Calculate the protrusion feature of point A_i and define the protrusion feature of point A_i as:

$$Atc_i = \left\| \overrightarrow{f_i} \right\|_2 \times e^{c_i}. \tag{11}$$

The protrusion feature comprehensively considers the average resultant vector and taper feature of the current point, where $\left\|\vec{f}_i\right\|_2$ is the penalty coefficient and has the function of highlighting taper features and suppressing planar features. In the case of relatively uniform local distribution of point clouds, for local areas with larger taper characteristics, such as the concave and convex points of objects, the larger the value, the closer plane $\left\|\vec{f}_i\right\|_2$ approaches 0. After calculating the *Atc* values of all points, compare the *Atc* values from both global and local perspectives to select the final key point. The algorithm flow for extracting key points is shown in Fig. 6.

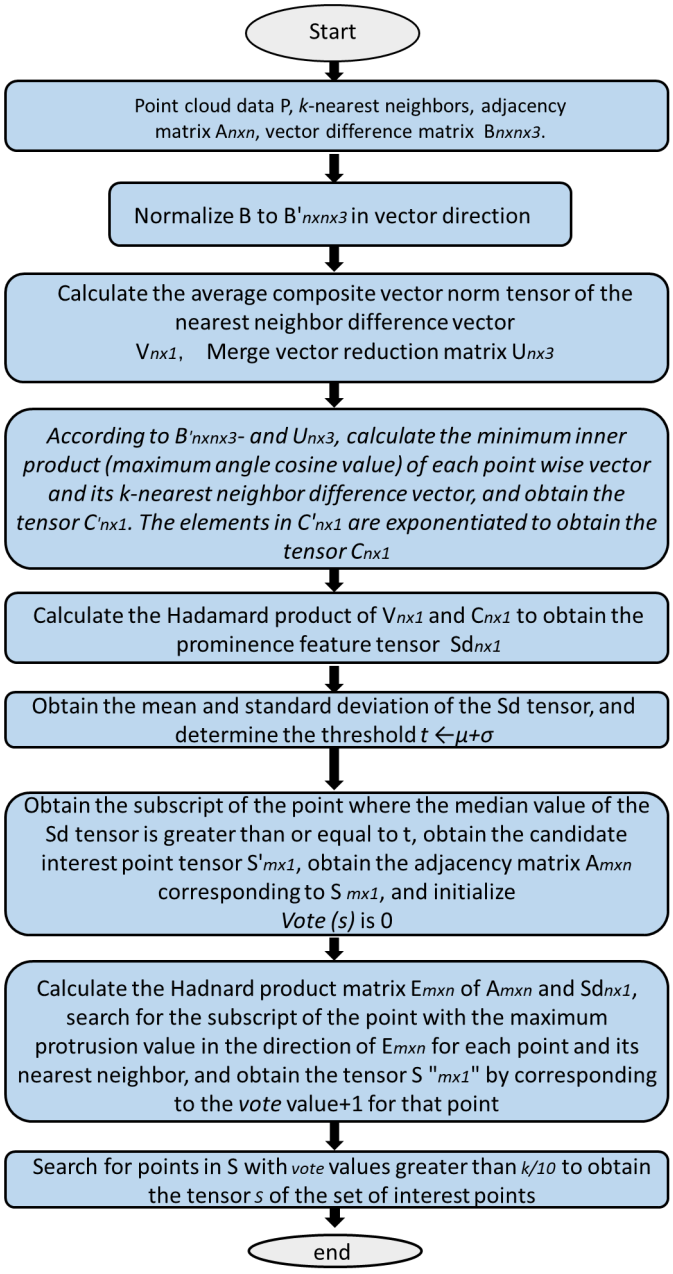

Fig. 6. Key point extraction algorithm process

4.3 Stacked Parts Identification

In real assembly scenarios, multiple parts are stacked, adhered, and obstructed, which makes it difficult to directly estimate the individual pose of the parts. The visual grasping of industrial robots requires real-time performance, and segmenting point cloud data in disordered scenes before performing pose calculation is a relatively efficient method.

This article uses the least squares method to segment the point cloud data obtained in the previous chapter. For any sampling point in point cloud A, search for its neighboring points through k–d-tree to form a subdomain P_d, and fit the surface within the subdomain. The expression for the fitting function is:

$$f(t)=\left[1,x,y,x^2,xy,y^2\right]^T\left[a_1(x,y),a_2(x,y),\cdots,a_m(x,y)\right]. \tag{12}$$

To solve the above fitting function $f(x)$, define the weighted sum of squares of the fitting function error H as the objective function, expressed as:

$$H=\sum_{j=1}^{n}\phi(h)\left[f(t)'-f(t)\right]^2. \tag{13}$$

$$\phi(h)=\begin{cases}e^{-\left(\frac{h}{\eta}\right)^2} & d\le 0.015\\ 0 & d>0.015.\end{cases} \tag{14}$$

In the formula, n represents the number of points within subdomain P_d; $\phi(h)$ represents the weight function, h represents the distance between the sampling point and its neighboring points, η is the coefficient related to the subdomain radius r, usually $\eta = 0.5r$. Use the above fitting formula to fit all points in the point cloud, reconstruct the point cloud surface based on the fitting function, obtain the smoothed 3D point cloud data, and then calculate the normal information of each point in the point cloud. The normal of any point in the point cloud is equivalent to the normal of the tangent plane fitted to the local area of the point cloud, so the problem of normal calculation can be transformed into a local plane fitting problem. Let the set of points on the plane to be fitted be:

$$A_O=\left\{(x_i,y_i,z_i),i=1,2,\cdots,n\right\}. \tag{15}$$

The distance from any point in the plane to the standard plane is expressed as:

$$d_i=\left|ax_i+by_i+cz_i-d\right|. \tag{16}$$

Among them, a, b, c are the plane coefficient, and the best fitting plane is solved using the principle of least squares, so that the sum of distances between the above points is minimized. Therefore, a Lagrangian function is constructed.

$$L(x)=\sum_{i=1}^{n}d_i^2-\lambda\left(a^2+b^2+d^2-1\right). \tag{17}$$

Take the partial derivatives of a, b, c, d in the equation separately, and record the results of each partial derivative as:

$$\Delta x_i=x_i-\frac{\sum_{i=1}^{n}x_i}{n}. \tag{18}$$

$$\Delta y_i = y_i - \frac{\sum_{i=1}^{n} y_i}{n}. \tag{19}$$

$$\Delta z_i = z_i - \frac{\sum_{i=1}^{n} z_i}{n}. \tag{20}$$

After obtaining partial derivatives, the Lagrange transform is ultimately represented using matrix equations, as follows:

$$\begin{bmatrix} \sum_{i=1}^{n} \Delta x_i \Delta x_i & \sum_{i=1}^{n} \Delta x_i \Delta y_i & \sum_{i=1}^{n} \Delta x_i \Delta z_i \\ \sum_{i=1}^{n} \Delta y_i \Delta x_i & \sum_{i=1}^{n} \Delta y_i \Delta y_i & \sum_{i=1}^{n} \Delta y_i \Delta z_i \\ \sum_{i=1}^{n} \Delta z_i \Delta x_i & \sum_{i=1}^{n} \Delta z_i \Delta y_i & \sum_{i=1}^{n} \Delta z_i \Delta z_i \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \gamma \begin{bmatrix} a \\ b \\ c \end{bmatrix}. \tag{21}$$

The final problem of solving the local surface normal vector of a point cloud is transformed into solving the eigenvectors and eigenvalues of the matrix in the above expression, and obtaining the eigenvector corresponding to the minimum eigenvalue as the main direction, that is, the direction of the normal vector of the point being solved. Calculate the normal information of the point cloud before and after smoothing using the above method, and visualize the target point cloud normal. Therefore, the segmented image after extracting point cloud features is shown in Fig. 7.

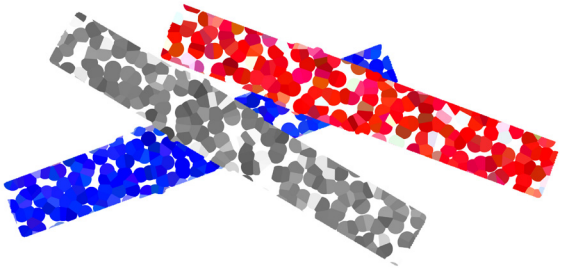

Fig. 7. Schematic diagram of point cloud segmentation effect

4.4 Pose Estimation

In the assembly process, the transmission speed of the production line is fast. Therefore, after point cloud segmentation of stacked parts, real-time and accurate pose estimation of each part is required before grasping. Therefore, this article adopts a multi task learning mechanism to establish an implicit connection between edge image reconstruction and pose estimation in the form of weight sharing. Under the guidance of edge reconstruction tasks, the pose estimation network will pay more attention to the feature information of the edge part in the image, thereby achieving higher accuracy and robustness in industrial part pose estimation [17]. The algorithm network is shown in Fig. 8.

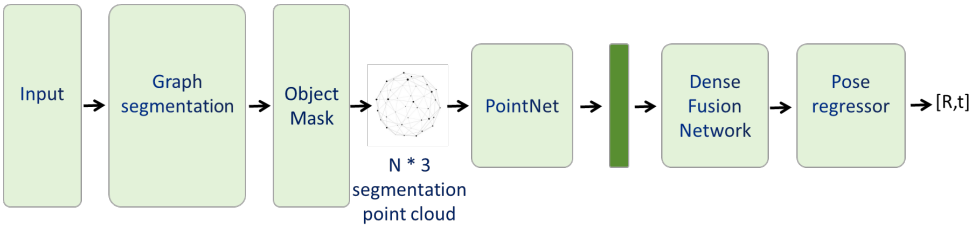

Fig. 8. Algorithm framework

After point cloud feature extraction and point cloud segmentation, the obtained point cloud data is sent to the PointNet [18] architecture to extract the geometric features of the point cloud. The network structure of PointNet directly acts on the point cloud, including the overall structure including T-net layer, convolution layer, pooling layer, and feature fusion layer. When the data volume of a point cloud is large, inputting all the data into the network at once not only occupies memory but also easily affects the computing speed. Therefore, the PointNet network itself includes a data normalization processing layer. In the segmentation part, the relative position of each point in the dataset to the corresponding room is represented by coordinates. Therefore, each point can be expressed as a 9-dimensional vector, which includes coordinate information, color information, depth information, and angle information. During the training process, each point in the block was sampled, and during testing, all points in the block were also tested using the k-fold training strategy. The specific operation process is as follows:

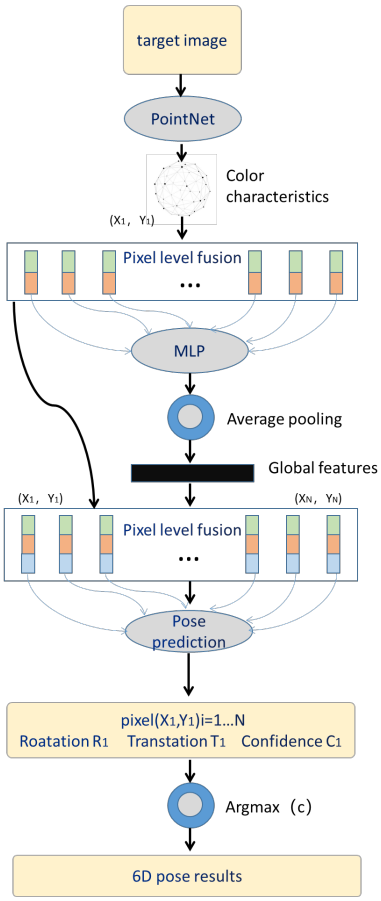

Fig. 9. Pose estimation flowchart

Randomly sample *NN* points on the object point cloud, where *N* points correspond to *N* pixel positions on the image. Thus, the correspondence between two-dimensional images and three-dimensional point clouds was obtained. Furthermore, a corresponding color embedding feature is fused into the feature vector of each point in the point cloud to obtain a richer feature expression. The above steps perform local feature extraction on a single point. In addition, the algorithm does not discard global features. It performs global average pooling on the fused color and point cloud features, that is, average pooling on each feature dimension of *N* points. In this way, a new feature vector is obtained, where each value of the feature vector represents the average value of all image points and point cloud points in this feature dimension. Subsequently, this "global feature" is also fused into each "local feature", resulting in pixel level fused features. Subsequently, the pose prediction module regresses the pose parameters. Among them, each densely fused feature predicts a set of object poses, that is, outputs a rotation, translation, and confidence for each point, and finally takes the rotation and translation parameters corresponding to the highest confidence value as the pose result. The pose estimation flowchart is shown in Fig. 9.

After the above process, the pose of stacked parts is estimated, and a neural network-based iterative optimization method is used to optimize the pose to improve grasping accuracy. The network optimization methods are as follows:

$$[R,t]_{youhua} = [R,t]_k \cdot [R,t]_{k-1} \cdots [R,t]_0 . \tag{22}$$

$[R,t]_k$ represents the transformation matrix obtained by the pose estimation network, and $[R,t]_{youhua}$ represents the final result after pose optimization. The pose optimization network structure is shown in Fig. 10.

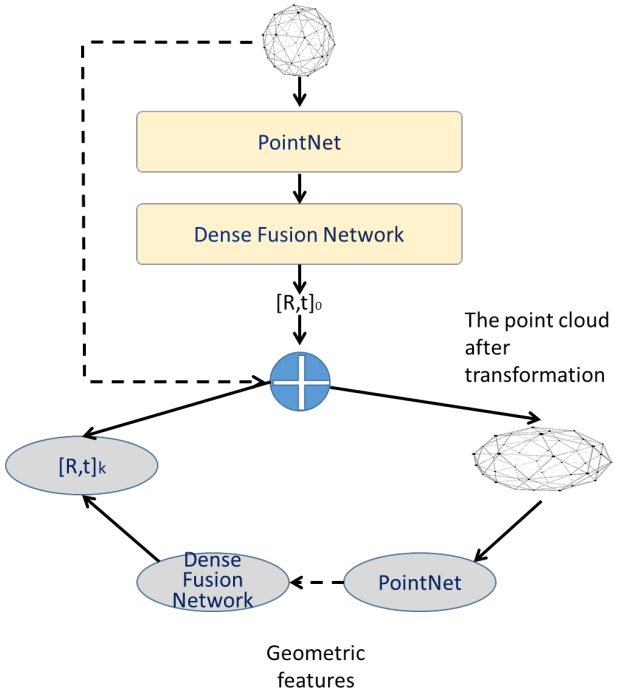

Fig. 10. Pose estimation flowchart

Therefore, the pseudocode for optimizing the entire algorithm process is as follows:

Solution steps

1: Input: scene image

2: Initial solution: Samples N points at the intersection of the Mask image and the Depth image, corresponding to X points in the point cloud *PP* 0 and X pixels in the RGB image;

3: Share the hourglass network of the first branch, encode the features of the segmented object RGB image, and obtain image features;

4: Using PointNet for feature encoding of point clouds to obtain geometric features;

5: Dense feature fusion is performed on image features and point cloud features pixel by pixel to obtain local point features; Global description of local point features to obtain global features;

6: Splicing local point features and global features;Using a pose regressor input, calculate a pose result for each point in the point cloud;

7: Take the pose result with the highest confidence as the initial pose result.

8: for $k \in n$

9: Transform the point cloud using pose to obtain the transformed point cloud;

10: Using networks to extract point cloud features from point clouds;

11: For segmented images, use the same color features based on the hourglass network of the second branch in step 3;
End for

11: Output: The pose result of the target object $[R, t]_{youhua}$.

In summary, after the above process, this article has completed the feature recognition of stacked parts. This section has completed the pose recognition work for stacked parts, and the accuracy of recognition determines the accuracy of grasping. Therefore, in this section, the following work has been mainly completed: point cloud feature extraction, point cloud feature point preprocessing, point cloud keypoint extraction, and detailed process description of four key steps in stacked part pose recognition. In the stacked part pose recognition, this paper presents a recognition model and designs pseudocode based on this model, providing more feasible reference examples for other researchers.

5 Simulation Experiments and Result Analysis

To fully verify the accuracy of the grasping method described in this article, experimental verification was conducted from two aspects: recognition accuracy and grasping accuracy.

5.1 Recognition Accuracy

Scatter and stack the target axes in sequence within the field of view of the grasping system. Firstly, use a depth camera to measure the 3D point cloud data of the parts, and then conduct a point cloud matching experiment, repeating a total of 100 times. After each successful algorithm registration and recognition of the part point cloud, a visualization result will be output, as shown in Fig. 11. Determine whether the recognition was successful and accurate based on the captured results. Record the number of successful attempts by the robot to grasp the target object, and obtain the statistical results of the system's grasping success rate, as shown in Table 3.

Table 3. System's grasping success rate

Axis number	Success times	Number of failures	Success rate
Axis 1	98	2	98%
Axis 2	99	1	99%
Axis 3	97	3	97%

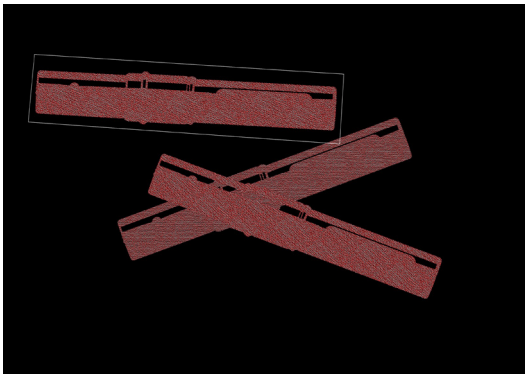

Fig. 11. Axis recognition results

5.2 Grab Accuracy Test

The hardware of a high-precision 3D disordered grasping system for industrial robots includes 3D vision measurement equipment, six joint robots, and specialized flexible fixtures. The system error is mainly composed of measurement error, hand eye calibration error, robot positioning error, and other comprehensive errors. The factors that affect the system error in each link are shown in Table 4

Table 4. System error in each link

Parameter	Parameter values
Measurement error	$\leq 0.77mm$
Hand eye calibration error	$\leq 0.46mm$
Robot positioning error	$\leq 0.04mm$

Therefore, the maximum theoretical error of the entire system is 0.77. Using the results calculated by this grasping system as the measurement value, 5 grasping accuracy experiments were conducted on randomly stacked parts. The experimental data of randomly stacked parts are shown in Fig. 12.

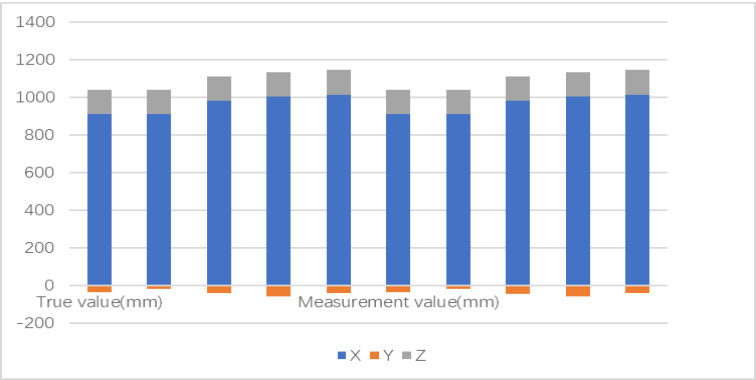

Fig. 12. Grab accuracy analysis

From the above figure, it can be seen that the average translation errors in directions x, y, z are 0.154mm, 0.08mm, -0.08mm, respectively. The maximum of the average translation errors in the three directions is used as the translation error of this system, and the maximum of the average angle error in the three aspects is used as the angle error of this system. Therefore, the proposed disordered grasping system has a translation error of 0.77mm,

which is smaller than the theoretical maximum error of the system and can meet the automatic recognition and positioning grasping requirements of scattered stacked parts.

6 Conclusion

3D machine vision has been widely applied in various fields such as high-altitude radar surveying, industrial part recognition, and medical image reconstruction. Point cloud data registration, as a key technology in 3D machine vision, directly affects the effectiveness of 3D object recognition and reconstruction. This article takes the stacking part picking technology in 3D machine vision as the research object, and proposes a registration algorithm based on artificial intelligence algorithm to address the problem of accuracy decline caused by prior assumptions in point cloud data registration. The algorithm dynamically optimizes the recognition parameters, enabling it to have the ability to match adaptively during the registration process, improving the registration accuracy of part point clouds, and verifying the effectiveness of the algorithm through experiments.

This article conducts in-depth research on particle swarm optimization based registration algorithms. Although some research results have been achieved, there are still some shortcomings that need to be explored and improved.

1) The camera calibration work was not carried out. Although the pose estimation of the target part in the scene was achieved based on feature point pairs, the transformation relationship between the pose of the target part, the camera coordinate system, and the world coordinate system was not carried out, and the relative relationship between the part coordinates and the robotic arm was not determined. In the next step of research work, it will be further improved.

2) The grasping of parts by the robotic arm has not been achieved. This type of work requires analyzing the geometric features of the parts, determining the grasping pose point of the target part, and synchronously calculating the corresponding optimal path of the robotic arm. In the next research work, the path planning of the robotic arm will be improved by combining the installation position of the robotic arm with the height of the workbench.

3) Further registration performance comparison experiments with other types of swarm intelligence algorithms are needed.

7 Acknowledgement

Research on industrial Robot grasping detection based on deep learning and machine vision. (2022ZC026).

References

[1] L.-H. Wu, J. Bai, G.-P. Kang, G.-C. Huang, Counting and Positioning Method of Threaded Steel Plate Welding Robot Based on Deep Learning and 3D Vision, Mechanical & Electrical Engineering Technology 52(12)(2023) 73-76.

[2] H.-B. Ma, X. Wang, J.-Y. Bian, Y. Jin, Review of 3D machine vision development and its industrial application, Micro / nano Electronics and Intelligent Manufacturing 4(4)(2022) 50-61.

[3] Y.-G. Liu, F.-N. Yu, X.-J. Zhang, Z. Chen, D.-T. Qin, Research on 3D Object Detection Based on Laser Point Cloud and Image Fusion, Journal of Mechanical Engineering 58(24)(2022) 289-299.

[4] H. Luo, J.-Y. Shang, R.-F. Li, J. Guo, N. Yang, C. Guo, A High-precision 3D Disordered Grabbing System for Industrial Robots, Navigation and Control 22(4)(2022) 106-116.

[5] P. Chen, C. Li, X.-J. Lei, Robotic assembly of cylindrical shaft and hole parts based on 3D vison, image detection and admittance control, Control and Decision 38(4)(2023) 963-970.

[6] Y.-F. Xu, W.-J. Wang, Z.-G. Wang, B. Chen, Rasping and Shaft Hole Assembly Based on Dual Arm Robot and 3D Vision, Small & Special Electrical Machines 50(1)(2022) 27-31.

[7] X.-D. Cui, H. Tan, P.-J. Wang, L.-B. Xu, J.-H. Chen, Research on vision − based robot's grabbing attitude for scattered stacking workpieces, Manufacturing Technology & Machine Tool (2)(2021) 36-41.

[8] J. Tao, Y.-C. Wu, X.-H. Zhu, H.-C. Yu, J.-J. Wang, X.-Y. Chen, Disorderly grasping parts of pump production line based on 3D vision, Journal of Mechanical & Electrical Engineering 39(5)(2022) 604-611+640.

[9] S.-Y. Zhu, B.-J. Xiao, S.-Q. Huang, Irregular Part Grasping System of Manipulator Based on 3D Vision, Automobile Technology & Material (4)(2023) 60-67.

[10] J.-W. Zhang, H.-R. Li, T.-H. Luo, L.-S. Zhang, Research on the external size measurement for industrial pipes using multi-group binocular vision system, Electronic Measurement Technology 46(6)(2023) 137-146.
[11] W.-S. Deng, Q. Liu, R.-L. Zhao, A Distributed Approximate Physical Integrated Commissioning Method for Production Lines Based on Digital Twins, Industrial Engineering Journal 26(5)(2023) 124-130.
[12] M. Guo, B.-J. Hou, J.-S. Gao, H.-H. Zhang, Research on Part Pose Measurement System Based on Binocular Vision, Machinery Design & Manufacture (9)(2023) 145-149.
[13] J.-Y. Zhang, S.-Q. Wu, B. Chen, Q. Zhang, L.-X. Liao, Binocular Vision Based Multi-dimensions On-line Measuring System for Workpieces, Instrument Technique and Sensor (10)(2018) 75-80.
[14] Z.-S. Liu, Y.-T. Fan, F. Chen, Research on recognition and positioning of stacked parts based on binocular vision, Transducer and Microsystem Technologies 42(2)(2023) 41-44.
[15] H.-T. Yang, G.-H. Lu, M. Shen, Research on accurate reconstruction method of gear 3D point cloud based on binocular vision*, Transducer and Microsystem Technologies 42(10)(2023) 56-59.
[16] H.-W. Wang, P. Guo, T.-B. Hang, T. Chen, Research on Extraction of Key Point Coordinates of Bridge Piers Based on UAV Close-Range Photogrammetry, Journal of Geomatics 48(6)(2023) 51-55.
[17] J.-J. Meng, X.-T. Chen, D.-C. Li, W.-Z. Qi, Pose Estimation Processing Method of Computer Vision Technology, Computer Simulation 40(5)(2023) 274-278.
[18] L.-L. Mu, Z.-J. Shan, Anti Interference Research of 3D Point Cloud Component Segmentation Based on PointNet, Journal of Suihua University 43(8)(2023) 144-147.