

## Article

# Research on Multi-Hole Localization Tracking Based on a Combination of Machine Vision and Deep Learning

Rong Hou <sup>1</sup>, Jianping Yin <sup>1,\*</sup>, Yanchen Liu <sup>1</sup> and Huijuan Lu <sup>2</sup>

<sup>1</sup> School of Mechanical and Electrical Engineering, North University of China, Taiyuan 030051, China; b20220106@st.nuc.edu.cn (R.H.); liuyanchenlaoliu@nuc.edu.cn (Y.L.)

<sup>2</sup> School of Life and Environmental Sciences, Guilin University of Electronic Technology, Guilin 541004, China; lh\_lhj0307@163.com

\* Correspondence: yjp123@nuc.edu.cn

**Abstract:** In the process of industrial production, manual assembly of workpieces exists with low efficiency and high intensity, and some of the assembly process of the human body has a certain degree of danger. At the same time, traditional machine learning algorithms are difficult to adapt to the complexity of the current industrial field environment; the change in the environment will greatly affect the accuracy of the robot's work. Therefore, this paper proposes a method based on the combination of machine vision and the YOLOv5 deep learning model to obtain the disk porous localization information, after coordinate mapping by the ROS communication control robotic arm work, in order to improve the anti-interference ability of the environment and work efficiency but also reduce the danger to the human body. The system utilizes a camera to collect real-time images of targets in complex environments and, then, trains and processes them for recognition such that coordinate localization information can be obtained. This information is converted into coordinates under the robot coordinate system through hand-eye calibration, and the robot is then controlled to complete multi-hole localization and tracking by means of communication between the upper and lower computers. The results show that there is a high accuracy in the training and testing of the target object, and the control accuracy of the robotic arm is also relatively high. The method has strong anti-interference to the complex environment of industry and exhibits a certain feasibility and effectiveness. It lays a foundation for achieving the automated installation of docking disk workpieces in industrial production and also provides a more favorable choice for the production and installation of the process of screw positioning needs.

**Keywords:** machine vision; deep learning; robotic arm; hand-eye calibration; ROS communications; porous disk



**Citation:** Hou, R.; Yin, J.; Liu, Y.; Lu, H. Research on Multi-Hole Localization Tracking Based on a Combination of Machine Vision and Deep Learning. *Sensors* **2024**, *24*, 984. <https://doi.org/10.3390/s24030984>

Academic Editors: Yexin Wang and Yi Cui

Received: 22 December 2023

Revised: 21 January 2024

Accepted: 31 January 2024

Published: 2 February 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

With increasing technological progress, it is inevitable that the automation of industrial production will become more widespread. However, simple industrial robots have been unable to meet the complexity, variability, and unstructured production forms characterizing modern industrial production environments. Machine vision systems use machines instead of human eyes to execute a variety of measurements and judgments. By such means, industrial robots are able to perceive their environments. The addition of industrial cameras to robots is equivalent to adding eyes to the robot, which is then able to recognize workpieces within the range of the camera lens [1–3]. Yang et al. [4] from Guizhou University have proposed an intelligent algorithm and system design based on machine vision to achieve accurate recognition and smooth tracking of specific dynamic targets. This system uses a Jetson TX2 as a control core and adopts a convolutional neural network based on the YOLOv3 algorithm to locally realize dynamic target recognition, which is combined with a tracking and obstacle-avoidance algorithm based on fuzzy logic. This approach was used

to determine the steering of the target movement and the rotation amplitude, as well as the positions of obstacles, resulting in precise control of the system's movement and effective obstacle avoidance. Lin et al. [5] have proposed a teleoperated human–computer system based on a six-degrees-of-freedom robotic arm. This system uses a KinectV1 camera, a UR3 robotic arm, and the Microsoft Skeletal Recognition Library to achieve recognition of basic human gestures, as well as real-time tracking of human arm movements by the robotic arm, by means of mapping between the joints of the human arm and those of the robotic arm. The Xinsong Robotics Company (Tianjin, China), in conjunction with the Shenyang Institute of Automation of the Chinese Academy of Sciences, have developed the SR10C-7H welding-specific robot [6], which was equipped with a fixed-machine vision CCD camera in an automated assembly environment to obtain spatial information on workpieces and feed this back to the robotic arm motion actuator through a vision processing system [7]. In Germany, the KUKA company (Augsburg, Germany) has developed a system based on vision and haptic perception for the BMW Mini assembly line; this involves the use of an IBR-iiwa-type industrial robotic arm and an ISRA vision system to achieve rapid recognition of target object shape contours and gripping positions through perception of the surrounding working environment, in order to determine the specific mounting position of workpieces [8]. Xiao et al. [9] have used a magnetic-resonance-compatible robot to increase the operating range of FUS applications and proposed an active tracking and detection method for a robotic arm based on a magnetorheological coordinate system to carry out a robotic-arm-assisted and magnetic-resonance-guided active tracking study of focused ultrasound ablation. Guo et al. [10] have proposed a machine-vision-based screw-hole localization detection method to address the problems of high intensity and low efficiency associated with the manual installation of workpieces. Industrial machine vision systems can replace the human eye to carry out tasks including detection, measurement, identification, and localization [11,12]. In addition, as such systems are able to overcome the inconsistencies associated with human eye standards, they can be customized to meet higher digital standards in industrial quality control and achieve performance levels beyond the limits of the human eye with respect to speed, spectrum recognition, resolution, sensitivity, and reliability. In short, the introduction of machine vision has greatly improved automation and intelligence on production lines [13,14]. Zhao [15] used the SIFT algorithm, Kalman filtering algorithm, and updating the size of the kernel window width to study the algorithm for target tracking of train hooks based on vision, and the results of the video tracking run concluded that the use of target tracking algorithms can be adapted to the working environment of the train-hook picking robot. However, the method is carried out on the basis of being able to recognize the target to be tracked, and the recognition of the target is affected by the actual complex environment, and this part will greatly affect the tracking of the target object, thus affecting the work efficiency. Zhao [16] used a mean shift based algorithm to realize the tracking of the workpiece by establishing a model of the target workpiece and comparing the similarity with the detection region and judging the similarity between the template region and the image of the workpiece to be processed by iterative computation, but the accuracy of this algorithm for the tracking of the workpiece is largely affected by the template features, as well as the light variations.

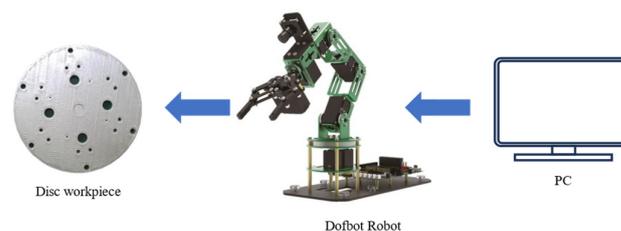
In industrial production, screw locking is an important process. When locking, it is necessary to accurately identify the position of each threaded hole such that phenomena such as leakage and floating locks may be avoided [17,18]. Traditionally, the positioning of threaded holes has been achieved by recognition with the human eye or by relying on fixtures. Both of these methods involve low efficiency and poor positioning accuracy, and both present difficulties when applied in rapid production [19]. Hadi et al. [20] evaluated the proposed AutoML model using the Case Western Reserve University (CWRU) bearing fault dataset by automated machine learning techniques, using ball bearing faults as an example, and achieved high accuracy, recall, precision, and F1 scores on the test and validation sets, which can help the related industries to reduce the time and cost associated with predictive maintenance. Also, the related work of Hadi et al. on AutoML predictive

maintenance fault classification study guided the conduct of the research in this paper. Therefore, in this paper, we consider the assembly process of a production line involving the automatic location and measurement of holes in a wide range of workpieces, as well as the problem that the accuracy of target detection and tracking is largely affected by the environment, and propose a multi-hole localization and tracking method based on machine vision and deep learning, which enables real-time tracking of disk workpieces in a complex on-site environment by classifying and detecting multiple circles on disk workpieces. Localization and tracking of workpieces with multiple holes can then be performed using a robotic arm, thereby improving the efficiency and reliability of the assembly. The primary contributions of this study are as follows:

1. Proposed a method of multi-hole localization and tracking by combining machine vision and deep learning, which reduces the impact of a single method;
2. The YOLOv5 network model is used for data training and detection, which improves the anti-interference ability of target recognition to the environment;
3. The communication between the upper and lower computer adopts ROS communication, which makes the control of the robot have a very good real-time performance;
4. The research ideas in this paper provide important insights for future researchers to improve the research when applied on a large scale.

## 2. Robotic Arm Multi-Hole Localization Tracking System

The robotic arm porous disk localization and tracking system proposed in this paper includes a PC, a robot, a camera, and a porous disk workpiece, as shown in Figure 1.



**Figure 1.** Schematic diagram of robotic arm multi-hole localization tracking system.

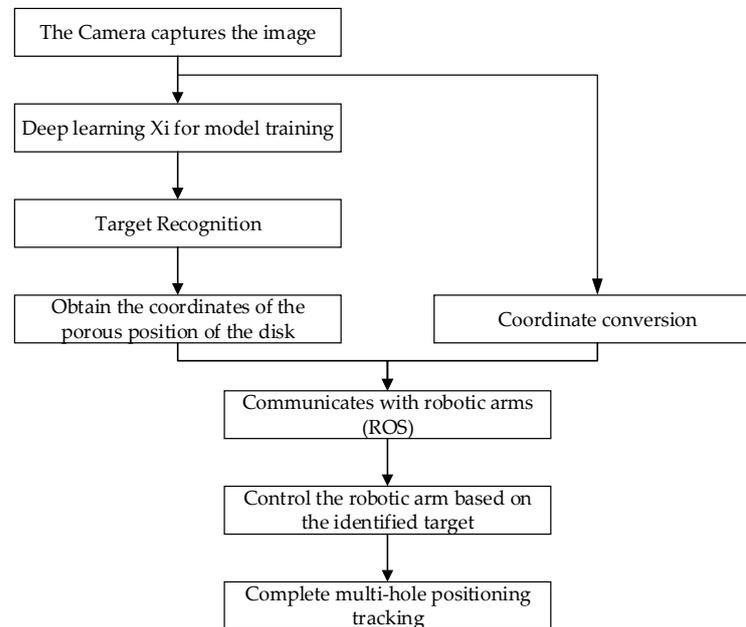
### 2.1. Working Principle of Porous Disk Positioning

Porous disk positioning is accomplished by means of automatic detection using a machine vision system that includes four links: image acquisition, image processing and data model training, system calibration, and communication between the upper and lower computers. The image acquisition link is responsible for acquiring images with the camera, obtaining high-quality images of the target area through a good distribution of light sources, and storing large numbers of images in the industrial control machine. The PC is responsible for image processing and data model training, processing the acquired images, extracting the eigenvalues of the multiple holes on the disk, obtaining the coordinate information of the multiple holes, and completing the tracking by adjusting the coordinates of the locating holes using the coordinate information of the four peripheral holes and the center hole. System calibration is carried out to convert the extracted coordinate information into the coordinate system of the robotic arm and to control the robotic arm to complete the real-time following of the disk workpiece. Communication between the upper and lower computers is mainly for the purpose of data communication between the PC and the robotic arm industrial control machine.

### 2.2. Localization and Tracking System Realization Process

The camera is fixed at the end of the robotic arm, the image information is acquired and transferred to the PC, and the model is trained on the PC. The acquired porous disk coordinates are first converted from 2D to 3D and are then converted to coordinates under the coordinate system of the robotic arm. The result of the conversion is used to control the

robotic arm movement through robot operating system (ROS) communication, such that the localization and tracking of the disk workpiece may be achieved. The flowchart of the tracking system is shown in Figure 2.



**Figure 2.** Flow chart of tracking system.

### 3. Deep-Learning-Based Target Recognition for Porous Disks

#### 3.1. YOLOv5 Deep Learning Framework

Considering the many sources of interference in real-world environments, we use the YOLOv5 (You Only Look Once version 5) deep learning framework to accurately capture target objects in the present study. YOLOv5 is a target detection model for recognizing and localizing objects in videos or images. It is characterized by its fast detection speed, high degree of accuracy, and good ability to detect small objects [21–23]. It is an end-to-end deep learning model that detects and localizes targets directly from raw images. It uses convolutional neural networks (CNNs) to learn the features of objects in an image and uses multi-scale prediction and grid segmentation to detect and localize targets. The advantage of YOLOv5 is that it is a single-stage target detection algorithm that works well with different image resolutions. In conclusion, YOLOv5 is an efficient target detection model that can be applied to many different scenarios, including autonomous driving, robot perception, and image analysis. Figure 3 shows a diagram of the core network structure of YOLOv5. Table 1 presents a comparison of model differences between different structures, where each ordinal number corresponds to the corresponding ordinal number in the network diagram of YOLOv5.

The number of convolutional kernels has a significant impact on the performance and effectiveness of the YOLOv5 model. A smaller number of convolutional kernels may lead to insufficient perceptual ability of the model to capture details and features in the image, thus affecting the accuracy of target detection. A larger number of convolutional kernels, on the other hand, may increase the computational complexity and the number of parameters of the model, leading to overfitting of the model or long training time.

In YOLOv5, the number of convolutional kernels is usually set by adjusting the width parameter of the network. A smaller width parameter will reduce the number of convolutional kernels, which is suitable for resource-constrained scenarios but may sacrifice some detection performance. A larger width parameter will increase the number of convolutional kernels and improve the model's perceptual ability and accuracy, but the computational and storage overhead will be also increased.

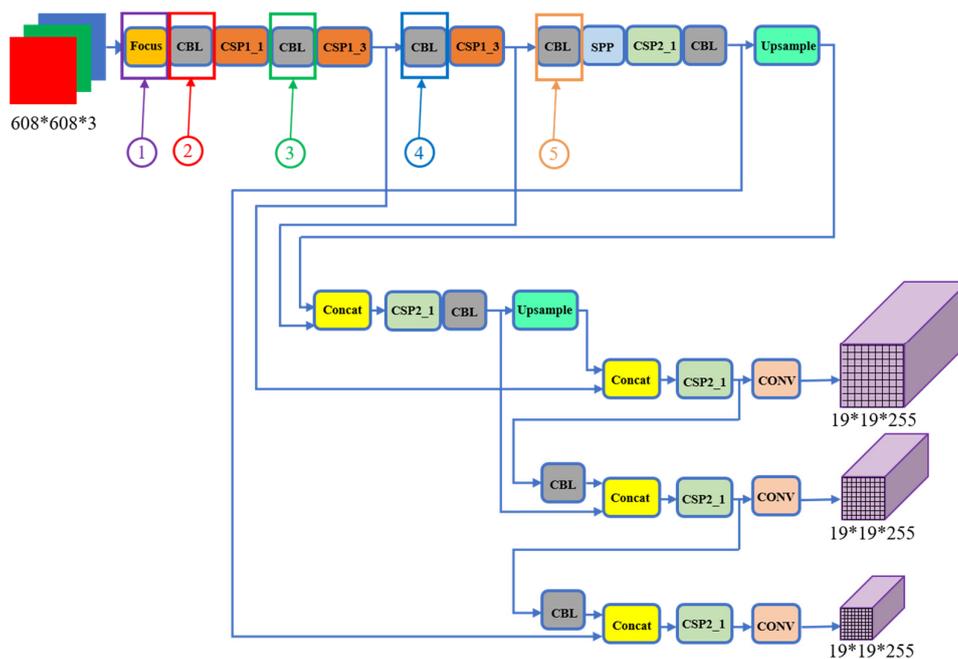


Figure 3. YOLOv5 core section.

Table 1. Comparison of network structure for different models of YOLOv5.

	YOLOv5s	YOLOv5m	YOLOv5l	YOLOv5x
① Number of convolution kernel	32	48	64	80
② Number of convolution kernel	64	96	128	160
③ Number of convolution kernel	128	192	256	320
④ Number of convolution kernel	256	384	512	640
⑤ Number of convolution kernel	512	768	1024	1280
Model depth multiple (depth_multiple)	0.33	0.67	1.0	1.33
Layer channel multiple (width_multiple)	0.5	0.75	1.0	1.25

Therefore, when using YOLOv5 for target detection, it is necessary to choose the appropriate number of convolutional kernels according to the specific application scenarios and resource constraints. Through experimentation and tuning, the optimal number of convolutional cores can be found to achieve a better detection effect and performance balance.

The four network structures are used to control the depth and width of the network through the parameters `depth_multiple` and `width_multiple` in Table 1. In particular, `depth_multiple` controls the depth of the network, while `width_multiple` controls the width of the network. Among these four networks, YOLOv5s has the smallest model and the fastest detection speed, but its level of accuracy is relatively low; YOLOv5m and YOLOv5l have successively larger models and higher levels of accuracy; and YOLOv5x has the highest level of accuracy, but also the slowest detection speed. As the targets used in this study were relatively small and, so, high accuracy was required, the YOLOv5x network structure was selected for training and detection.

### 3.2. Model Training

Considering the subsequent development and research, the camera selected in this paper is the Realsense D415 camera, which has a high resolution and can measure the depth (the minimum depth can be measured to 35 cm). Since the overall size of the robot arm in this paper is less than 35 cm, no research has been added to this part. Therefore, images with a resolution of  $640 \times 480$  were collected for training. After a certain number of images (about 2000) were

collected by the camera, a dataset suitable for YOLOv5 training was generated by labeling the targets of the images individually, for the purposes of training and testing. The precision of object detection is mainly measured by precision, recall, F1 Score, PR\_curve, mAP, and other indexes. Figures 4–6 show the indicators after model training, and the recognition accuracy of the target object is measured by analyzing relevant indicators.

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$F1Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (3)$$

where *TP*: true positive; *FP*: false positive; *FN*: false negative. And the value of *F1* ranges between 0 and 1, with 1 being the best and 0 being the worst.

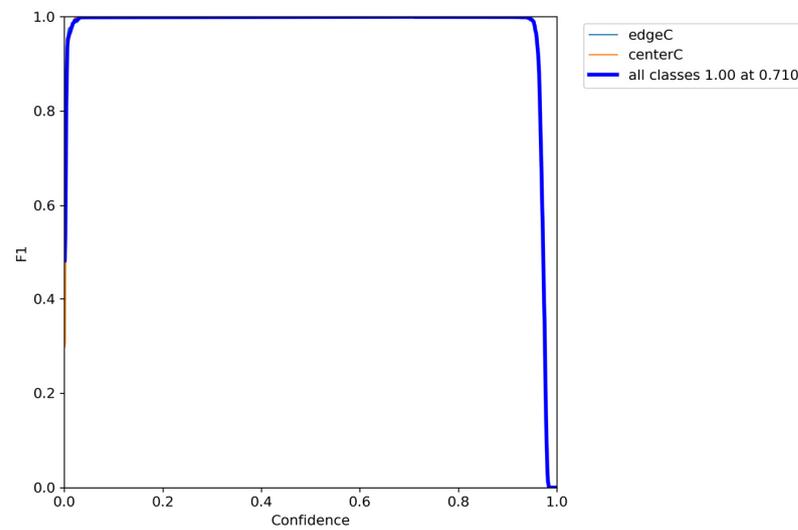


Figure 4. F1\_confidence curve.

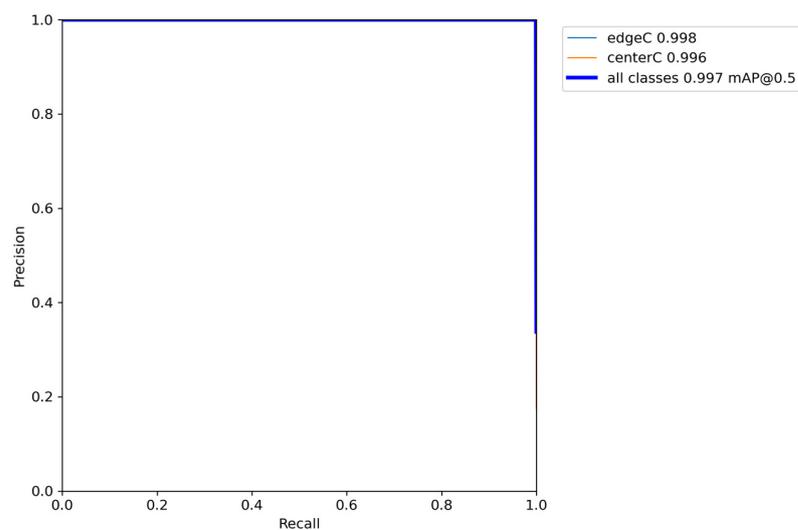
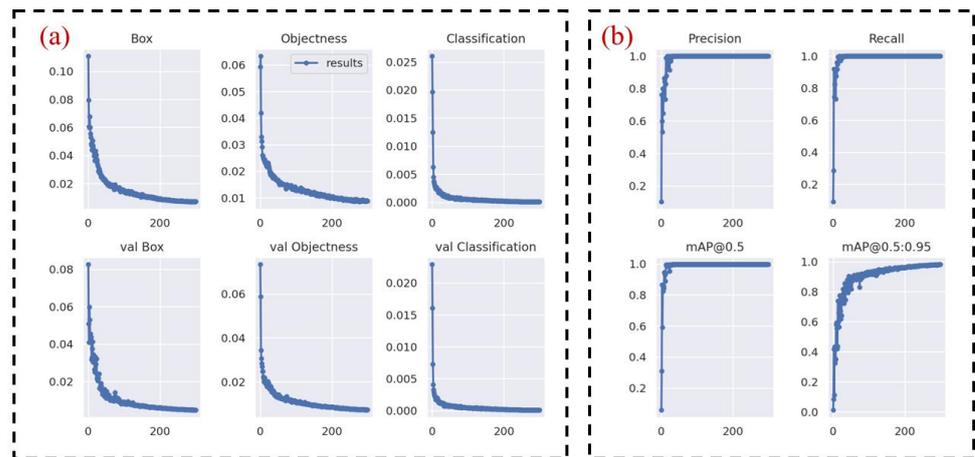


Figure 5. PR\_curve.



**Figure 6.** Training each evaluation index: (a) training evaluation index; (b) detection accuracy of the target.

For the PR\_curve in Figure 5, the PR\_curve is the relationship curve between precision and recall, where P is mAP (mean average precision). It is expected that both P and R can obtain 1 under the premise of high accuracy. We sought to detect all categories with the highest possible degree of accuracy; therefore, when the curve is close to the point (1, 1)—that is, when the area of the mAP curve is close to 1—the results are better. As can be seen from the figure, the area of the mAP curve trained for the purposes of the present study was almost 1.

The loss function is used to measure the degree of difference between the predicted value and the real value of the model, which largely determines the performance of the model.

For each training evaluation index in Figure 6, the horizontal coordinate represents the number of training rounds, and for the six indexes in Figure 6a, it reflects the relationship between the presumed and verified mean value of the loss function. The smaller the mean value of the loss function in the vertical coordinate, the more accurate the target detection is, and the loss in the figure decreases to 0, so the training effect is better. For the four indicators in Figure 6b, they are the indexes to measure the quality of the detection accuracy of the target. The fluctuation of the accuracy and recall rate reflects the training effect. The smaller the fluctuation, the better the effect. It can be seen that the fluctuation in the figure is not large, so the training effect is better.

The YOLOv5 object detection pseudocode is as Algorithm 1:

---

**Algorithm 1.** The YOLOv5 object detection pseudocode

---

Input: images

Output: target detection information

1. Load pre-trained YOLOv5x model: `yolov5x.load_pretrained_model()`
  2. Define image preprocessing steps: `transform = transforms.Compose ([ transforms.ToTensor()])`
  3. Input images: `image = transform(Image.open(image_path)).unsqueeze(0)`
  4. Input the image into the YOLOv5x model:
 

```
with torch.no_grad():
    model.eval()
    predictions = model(image)
```
  5. Analysis of prediction results:
 

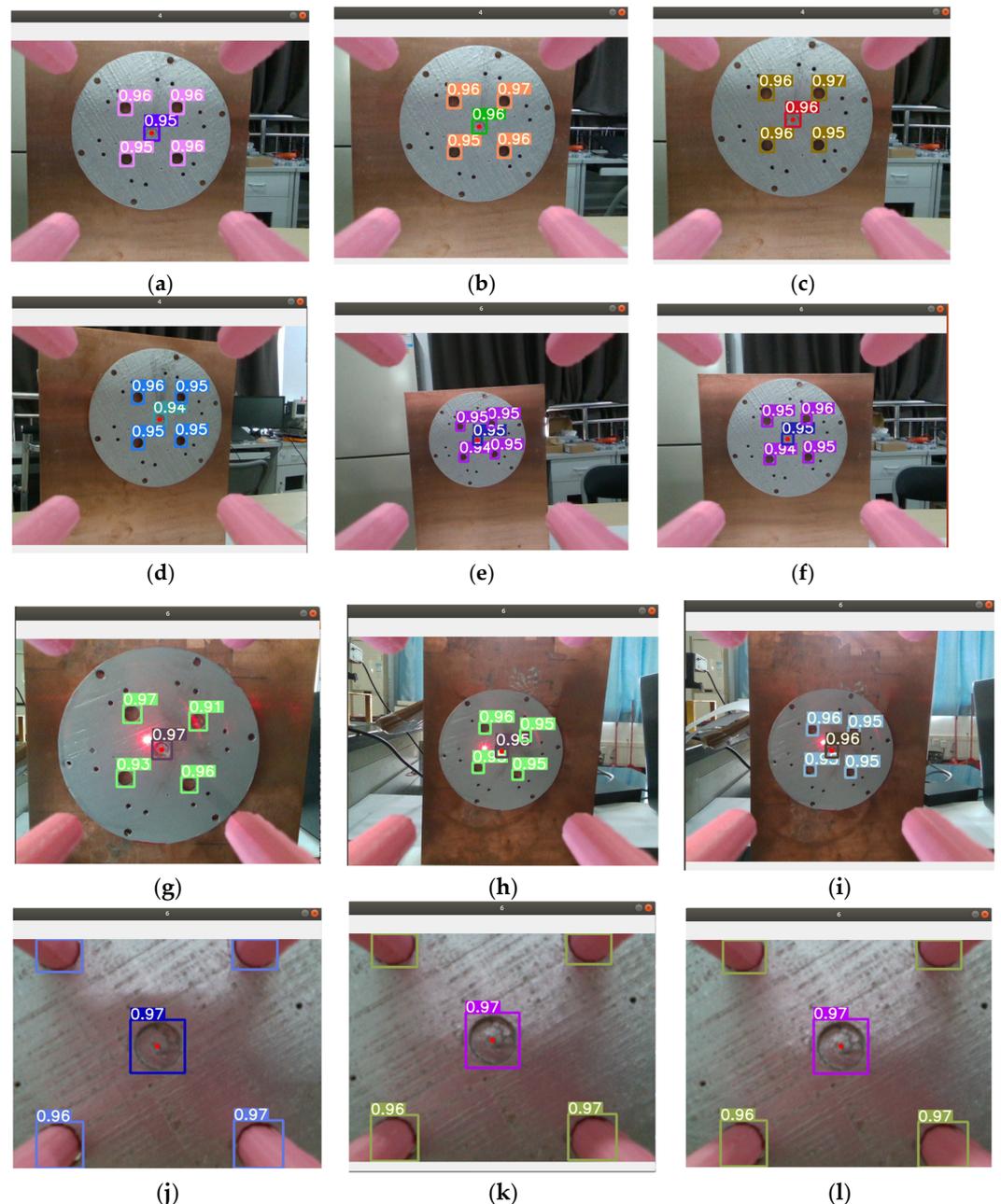
```
boxes = predictions[... ,:4] # Extract the bounding box coordinate information
scores = predictions[... ,4] # Extract the confidence
class_probs = predictions[... ,5:] # Extraction class probability
```
  6. Non-maximum suppression (NMS):
 

```
Selected_boxes = torch.ops.torchvision.Nms (boxes, scores,
iou_threshold = 0.5)
```
  7. Return the final result:
 

```
final_boxes = boxes[selected_boxes]
final_scores = scores[selected_boxes]
final_class_probs = class_probs[selected_boxes]
```
-

### 3.3. Target Recognition Detection

After training the dataset marked with labeling by the YOLOv5x network structure, we used the best weights obtained during training to carry out detection of porous disk workcraft and obtain the porous disk pixel coordinates. The identification and detection results are shown in Figure 7, from which it can be seen that the confidence of recognition reached a level above 90%.

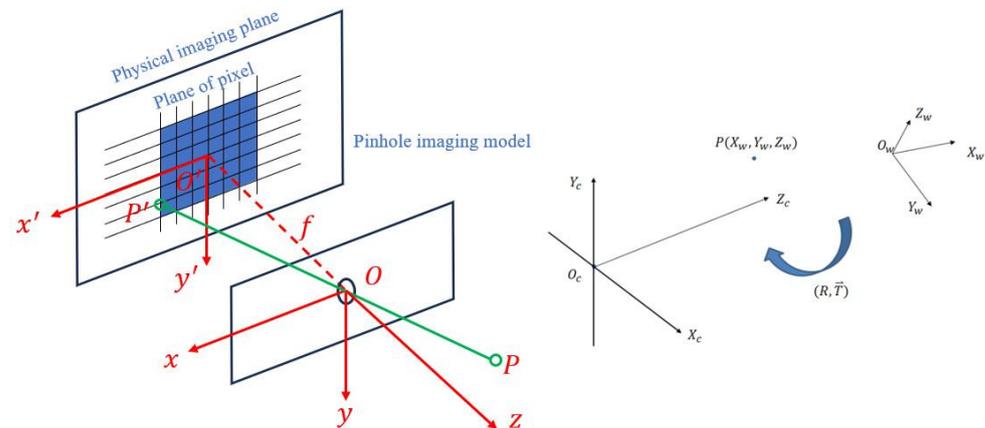


**Figure 7.** Training each evaluation index: (a–l) are the results of tests conducted at a distance; (j–l) are the results of the test at the closer location.

### 4. Coordinate Transformation and Calibration

Camera imaging is the mapping of real-world objects to an imaging plane. This principle is used for small-hole imaging models [24], an example of which is shown in Figure 8. As the coordinate information obtained by image processing is in the form of

pixel coordinates, in order to perform target tracking operations, the pixel coordinates are converted into world coordinates.

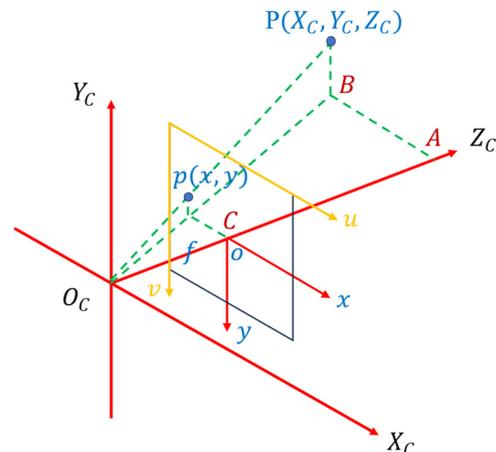


**Figure 8.** Camera imaging model.

#### 4.1. Coordinate Conversion

The world coordinate system is based on the three-dimensional spatial coordinates of the real world. It is therefore a three-dimensional coordinate system, as indicated by  $O_W - X_W Y_W Z_W$  in Figure 8.

The camera coordinate system is based on the principle of lens imaging and involves the presentation of world coordinates to the camera. It is also a three-dimensional coordinate system, as indicated by  $O_C - X_C Y_C Z_C$  in Figure 9.



**Figure 9.** Schematic diagram of camera and image conversion.

The image coordinate system is projected onto the screen, and the new coordinate system established does not include high-profile information. It is a two-dimensional coordinate system, as indicated by  $o - xy$  in Figure 9.

Pixel coordinates are pictures formed by discrete sampling of projection image coordinates. They are a two-dimensional coordinate system, as indicated by  $uv$  in Figure 9.

The conversion of world coordinates to pixel coordinates involves three transformations [25–27], as follows: conversion from world coordinates to camera coordinates, conversion from camera coordinates to image coordinates, and conversion of pixel coordinates to world coordinates. The conversion relationships for each coordinate system may now be stated.

The conversion between world coordinates and camera coordinates is as follows:

The coordinates of point P in the camera coordinate system may now be obtained as follows:

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix} = R \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} + T \Rightarrow \begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ \vec{0} & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (4)$$

where  $R$  is a  $3 \times 3$  matrix and  $T$  is a  $3 \times 1$  matrix.

The conversion between the camera coordinate system and the image coordinate system is as follows:

The relationship between the two groups of similar triangles,  $\Delta ABO_C \sim \Delta oCO_C$  and  $\Delta PBO_C \sim \Delta pCO_C$ , can be obtained as follows:

$$Z_C \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix}. \quad (5)$$

The conversion between pixel coordinates and world coordinates is carried out as follows:

$$Z_C \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & u_0 \\ 0 & \frac{1}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ \vec{0} & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ \vec{0} & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}, \quad (6)$$

where  $\begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$  is the internal reference of the camera and  $\begin{bmatrix} R & T \\ \vec{0} & 1 \end{bmatrix}$  is the external parameters of the camera.

$$\text{Equations (5) and (6) may now be deduced from } Z_C \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = M_C \begin{bmatrix} R & T \\ \vec{0} & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}:$$

$$Z_C M_C^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = RT, \quad (7)$$

$$\begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = R^{-1} (M_C^{-1} Z_C \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} - T), \quad (8)$$

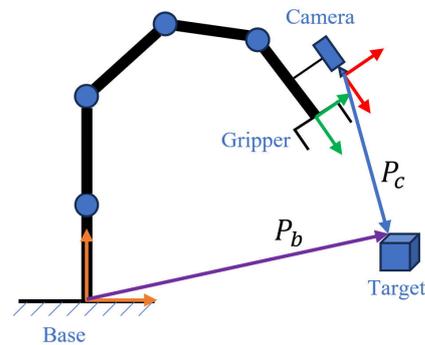
where  $M_C$  is the camera's intra-insinuity matrix.

After obtaining the internal and external parameters of the camera, the world coordinates of the object are derived by working backward through the relationship between pixel coordinates and world coordinates.

#### 4.2. Calibration Method

Camera calibration means that objects captured by the camera are in a three-dimensional world coordinate system. Camera imaging converts the three-dimensional camera coordinate system to the two-dimensional image coordinate system. Calibration allows for an approximate estimation of the transformation matrix and distortion coefficient. In unstructured and dynamic scenarios, hand-eye calibration is required for perception and localization [7–10]. The purpose of hand-eye calibration is to connect the perception module with the planning module of the robotic arm, such that the robotic arm can dynamically localize to the target position for grasping through the “eye”. The eye of the hand-eye

calibration is on the hand, so that “eye in hand” refers to the camera fixed to the end of the robotic arm, as shown in Figure 10.



**Figure 10.** Schematic of the “eye in hand” of a robotic arm.

The hand–eye calibration coordinate system is represented by four coordinate systems: The base coordinate system of the robotic arm, the end coordinate system of the robotic arm (end), the camera coordinate system (camera), and the calibration plate coordinate system (board).

The eye mainly calibrates the transformation matrix of the end of the camera and the robotic arm—that is, the transformation matrix  ${}_{end}^{camera}M$  from the coordinate system at the end of the robotic arm to the camera coordinate system. The implementation method is as follows:

1. The calibration plate is secured in a fixed position, such that it does not move;
2. The end of the robotic arm is moved to take multiple pictures of the calibration plate from different angles.

It is known that each image may be expressed as follows:

$${}_{end}^{camera}M = {}_{board}^{camera}M \times {}_{base}^{board}M \times {}_{end}^{base}M, \quad (9)$$

where  ${}_{board}^{camera}M$ , the calibration plate-to-camera transformation matrix (i.e., target pixel coordinates to camera coordinates), is obtained by solving for the captured calibration plate picture;  ${}_{end}^{base}M$  is obtained from the end position parameter of the robot arm; and  ${}_{end}^{board}M$  is an unknown quantity. This transformation matrix is the same for each set of pictures as the calibration plate is fixed in one position and does not move at any time in the process.

The deformation can now be expressed as:

$${}_{base}^{board}M = {}_{board}^{camera}M^{-1} \times {}_{end}^{camera}M \times {}_{end}^{base}M^{-1}. \quad (10)$$

From this, we obtain:

$${}_{board}^{camera}M_2 \times {}_{board}^{camera}M_1^{-1} \times {}_{end}^{camera}M = {}_{end}^{camera}M \times {}_{end}^{base}M_2^{-1} \times {}_{end}^{base}M_1. \quad (11)$$

Solving the system of equations  $AX = XB$  gives a value of  $X$  as the transformation matrix  ${}_{end}^{camera}M$  from the end of the robotic arm to the camera coordinate system, where

$$A = {}_{board}^{camera}M_2 \times {}_{board}^{camera}M_1^{-1}, \quad B = {}_{end}^{base}M_2^{-1} \times {}_{end}^{base}M_1. \quad (12)$$

Grasping of the target after hand–eye calibration may now be expressed as:

$$P_b = {}_{end}^{base}M \times {}_{end}^{camera}M \times P_c, \quad (13)$$

where  $P_c$  represents the vector of the clamped target in the camera coordinate system, which is calculated from the depth information and the pinhole camera model, and  $P_b$

represents the vector of the target in the basic coordinate system. The inverse kinematics of the manipulator can now be called on through  $P_b$  to actually complete the grasping and positioning tasks.

#### 4.3. Coordinate Mapping Reasoning

Coordinate mapping of each node was deduced for the six-degree-of-freedom robotic arm, and the analysis diagram is shown as Figure 11.

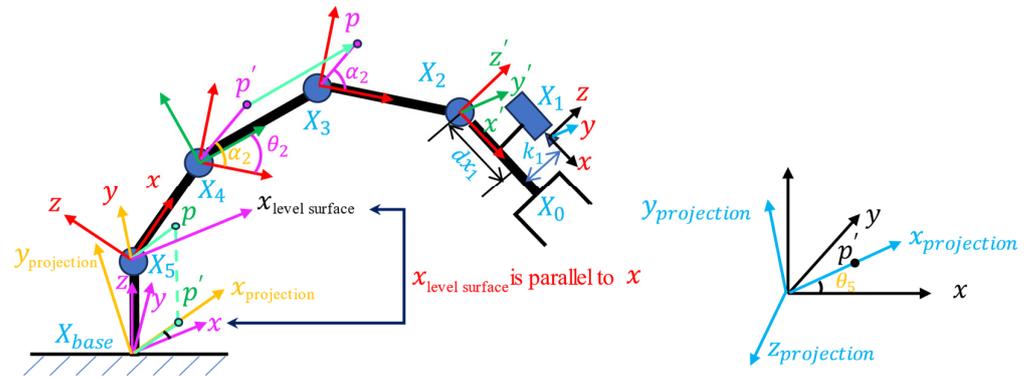


Figure 11. Coordinate mapping of each node.

Camera to end:

$$L_0 = (dx_0, dy_0, dz_0)^T \quad (14)$$

$$X_1 = X_0 + L_0 \quad (15)$$

where  $dy_0 = 0$ .

Camera to first joint:

$$L_1 = (dx_1, dy_1, dz_1)^T \quad (16)$$

where  $dy_1 = 0$ .

First joint coordinate system:

$$X_2 = X_1 + L_1 \quad (17)$$

$$X_2 = X_0 + L_1 = (dx_0 + dx_1, 0, dz_0 + dz_1)^T \quad (18)$$

First joint to second joint (rotation around the y-axis, y is constant, and the y-axis is omitted from the figure):

$$M_0 = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 \\ 0 & 1 & 0 \\ -\sin\theta_1 & 0 & \cos\theta_1 \end{bmatrix} \quad (19)$$

$$L_2 = (l_1, 0, 0)^T \quad (20)$$

where  $l_1$  is the rod length between the first and second joints.

Second joint coordinate system:

$$X_3 = M_0 X_2 + L_2 \quad (21)$$

$$X_3 = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 \\ 0 & 1 & 0 \\ -\sin\theta_1 & 0 & \cos\theta_1 \end{bmatrix} X_2 + L_2 \quad (22)$$

$$X_3 = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 \\ 0 & 1 & 0 \\ -\sin\theta_1 & 0 & \cos\theta_1 \end{bmatrix} \begin{bmatrix} x_0 + dx_0 + dx_1 \\ y_0 \\ z_0 + dz_0 + dz_1 \end{bmatrix} + \begin{bmatrix} l_1 \\ 0 \\ 0 \end{bmatrix} \quad (23)$$

$$M_1 = \begin{bmatrix} \cos\theta_2 & 0 & \sin\theta_{21} \\ 0 & 1 & 0 \\ -\sin\theta_2 & 0 & \cos\theta_2 \end{bmatrix} \quad (24)$$

$$L_3 = (l_2, 0, 0)^T \quad (25)$$

Third joint coordinate system:

$$X_4 = M_1 X_3 + L_3 \quad (26)$$

Fourth joint coordinate system:

$$X_5 = M_2 X_4 + L_4 \quad (27)$$

Variables:  $\theta_1, \theta_2, \theta_3, X_0$ , where  $\theta_1$  is angle difference between the first and second joints.

Base coordinate system:

(1) Projection:

Initial angle of the base joint:  $\theta_4$

Angle of point P in the  $X_5$  coordinate system with respect to the x-axis:  $\alpha_4$

Angle of connecting rod to the horizontal in joints III and IV:  $\alpha_{level\ surface}$

$$L_5 = (0, 0, z_5 + l_5)^T \quad (28)$$

$$X_{projection} = \begin{bmatrix} \cos\alpha_{level\ surface} & 0 & \sin\alpha_{level\ surface} \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \times X_5 \quad (29)$$

Variables:  $\theta_1, \theta_2, \theta_3, X_0, \alpha_{level\ surface}$ .

(2) Rotate  $\theta_4$  around the z-axis:

$$X_{base} = \begin{bmatrix} \cos\theta_4 & \sin\theta_4 & 0 \\ -\sin\theta_4 & \cos\theta_4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times X_{projection} + L_5 \quad (30)$$

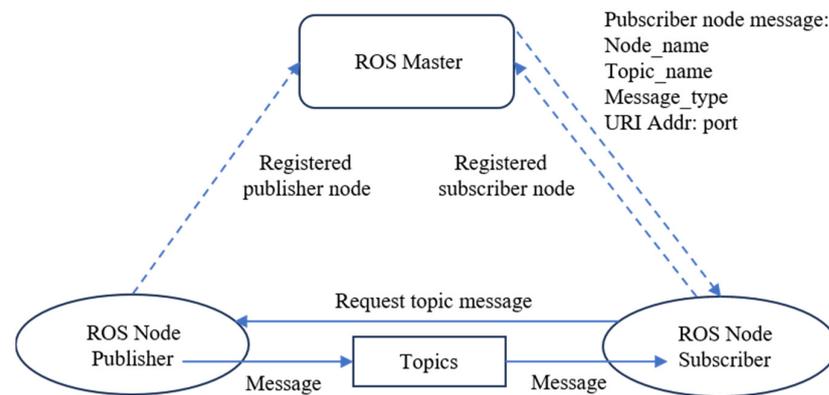
Variables:  $\theta_1, \theta_2, \theta_3, \theta_4, X_0, \alpha_{level\ surface}$ .

The relation obtained by the final inference is the relation of the calibration method described in the previous section.

## 5. Communication and Control of the Host Computer and Robotic Arm

The coordinate information obtained from the image processing part needs to be transferred to the robotic arm for control. In the present study, a topic communication mechanism was used to complete communication between the host computer and the robotic arm. Topic communication is one of the most commonly used communication methods based on the publish–subscribe model in ROS communication, which is mostly used in real-time cyclic data transmission scenarios involving less logic processing. The topic communication model is shown in Figure 12.

In the present study, this form was used to complete data transmission for real-time adjustment and positioning. In the following section, the direct communication between the ROS and the real robotic arm, as well as the control achieved, is specifically introduced.

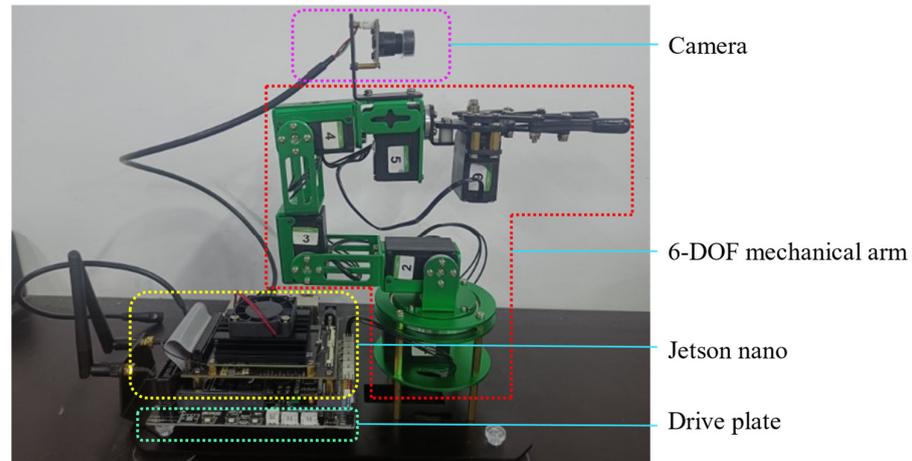


**Figure 12.** ROS topic communication model.

### 5.1. Robotic Arm Hardware Configuration

The joints of the Dofbot robotic arm use the DS-SY15A bus servo manufactured by De Sheng Model Technology Co., Ltd., located in Guangdong, China, as a power source, which is controlled by serial commands. The development board leads an IO port to connect to the servo, and the servo can be connected in series through the dual interfaces. This reduces the occupation of the development board's interface and enables reading of the angle, temperature, and voltage.

The servos have built-in driver circuits that send the correct data frames to the servos through the serial bus, such that control of servo rotation may be achieved. Six servos are connected to each other in series through the serial bus to achieve communication with the development board. The overall schematic of the Dofbot robotic arm is shown in Figure 13.



**Figure 13.** Schematic diagram of the robotic arm.

The Dofbot arm is driven by a multi-functional driver board. The driver board houses a Jetson Nano master control that runs Linux, which calls on the ROS MoveIt package for path planning and sends control commands to the driver board to make the six-degrees-of-freedom robotic arm move according to the desired path.

### 5.2. Control Algorithms

#### 5.2.1. Robot Kinematics

The robotic arm used in this paper is the Dofbot robotic arm, whose overall dimensions are shown in Figure 14 below:

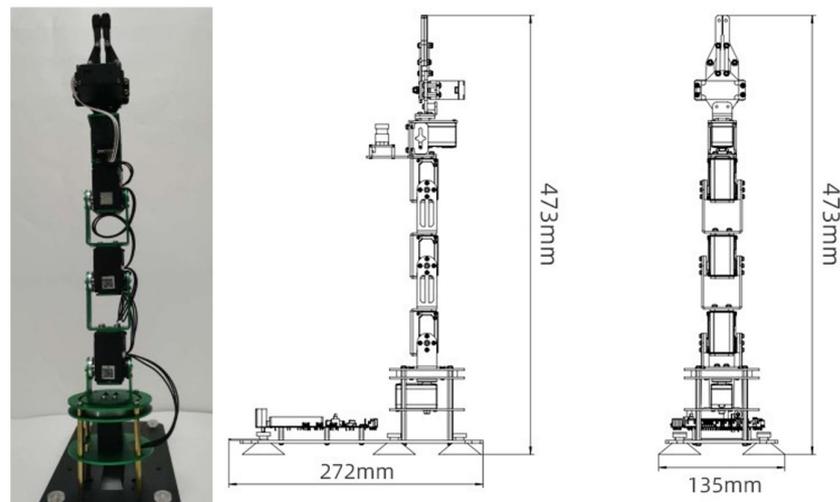


Figure 14. Dimensional drawing of robotic arm.

Dofbot's coordinate system is established as Figure 15:

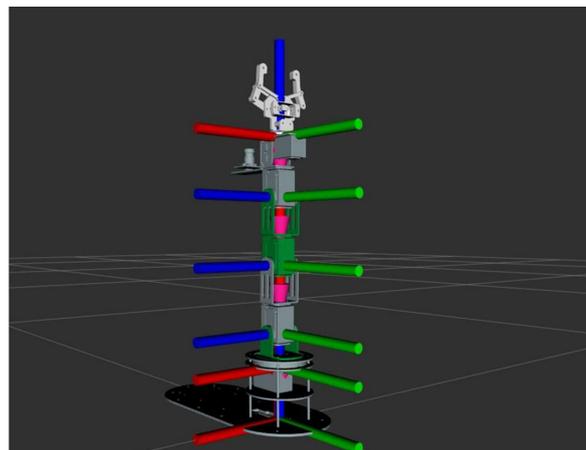


Figure 15. The picture of each coordinate system of Dofbot.

The red axis is the x-axis, the green axis is the y-axis, and the blue axis is the z-axis, where the bottom coordinate system is the base coordinate system.

$\alpha_{i-1}$ : The angle between  $\hat{Z}_{i-1}$  and  $\hat{X}_i$  as viewed in the direction  $\hat{X}_{i-1}$ ;

$a_{i-1}$ : The distance between  $\hat{Z}_{i-1}$  and  $\hat{X}_i$  as viewed in the direction  $\hat{X}_{i-1}$  ( $a_i > 0$ );

$\theta_i$ : The angle between  $\hat{X}_{i-1}$  and  $\hat{X}_i$  as viewed in the direction  $\hat{Z}_i$ ;

$d_i$ : The distance between  $\hat{X}_{i-1}$  and  $\hat{X}_i$  as viewed in the direction  $\hat{Z}_i$ ;

In this paper, the D-H parameter method is used to describe the geometric relationship between the connecting rod and the joint in the serial link of Dofbot manipulator. The D-H parameters of Dofbot are show as Table 2.

Table 2. D-H parameters of Dofbot.

	$d_i$ (Along the z-axis)	$a_i$ (Along the x-axis)	$\beta_i$ (Around y-axis)	$\theta_i$ (Around z-axis)
1	0.066	0	0	$\theta_1$
2	0.04145	0	$\frac{\pi}{2}$	$\theta_2$
3	0	-0.08285	0	$\theta_3$
4	0	-0.08285	0	$\theta_4$
5	0	-0.07385	$-\frac{\pi}{2}$	$\theta_5$

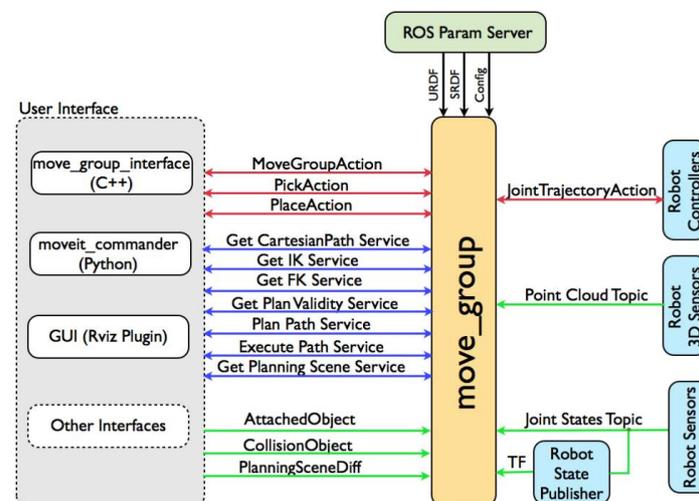
By solving the inverse kinematics of the robotic arm and analyzing its inverse kinematics, control of the robotic arm is achieved. Inverse kinematics refers to solving the angles of each rotational joint of a given robotic arm's end pose, which is a homogeneous transformation matrix.

The `dofbot.dofbot_getIK(urdf_file, targetXYZ, targetRPY, outjoints)` function is used for inverse kinematics calculation of Dofbot, which calculates the angles of each joint reaching the target point, where

- @param `urdf_file`: model file path;
- @param `targetXYZ`: target position;
- @param `targetRPY`: target attitude;
- @param `outjoints`: target point joint angle.

### 5.2.2. MoveIt Guides Path Planning

The MoveIt function package was used for path planning in the ROS robotics framework. The key node in MoveIt is `move_group`; its system structure is shown in Figure 16.



**Figure 16.** Move\_group node system structure.

The user provides endpoint target information to `move_group` through `move_group_interface`. Then, `move_group` carries out the path planning and releases the spatial trajectory from the waypoint information of each joint outward through `JointTrajectoryAction`; the robotic arm motion control node receives the information from `JointTrajectoryAction`, then controls the robotic arm's various joints to carry out the motion accordingly.

## 5.3. Realization Process

### 5.3.1. Secondary Development of Dofbot

Based on the `urdf` model of the Dofbot robot arm, a MoveIt function package was generated using the MoveIt configuration assistant for path planning.

### 5.3.2. Robotic Arm Control Node

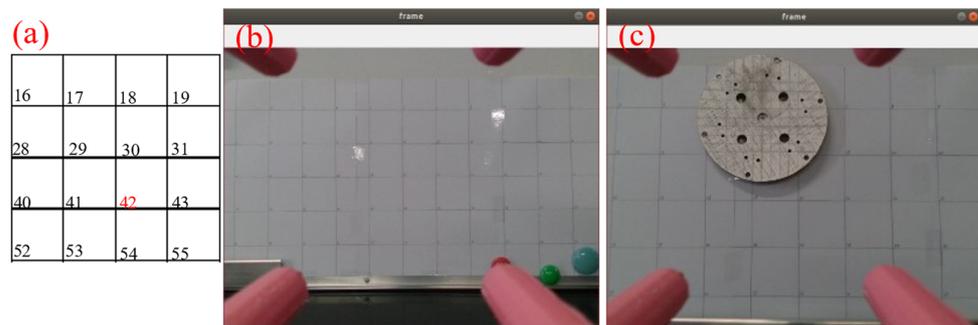
A robotic arm control node, `armCtrl`, was created to receive waypoints published by the `move_group` node, in order to control the movement of the robotic arm and to read back the current state of the robotic arm. When a valid `JointTrajectoryAction` is received, the node sends control commands to the robotic arm according to the path planned by `move_group` at a frequency of 10 Hz, controlling the robotic arm to perform a specific movement through the interface in the `Arm_Lib` class. At the same time, the `armCtrl` node reads the position information for each joint of the robotic arm at a frequency of 12.5 Hz, and publishes it with the topic "Joint States Topic". The `move_group` node can subscribe

to this topic to monitor the current position of the robotic arm in real time and realize closed-loop control.

## 6. Experimental Results

### Tracking Porous Disk Experiments

After a small laser module is installed above the camera at a horizontal distance of 0.25 cm and a vertical distance of 1.7 cm from the midpoint, a porous disk with a diameter of 14 cm is moved on the magnetic plate with grid lines (each grid is 5 cm) in the neighborhood of point 42 (see Figure 17a) for the numbering of each neighborhood to carry out target tracking and detection. The following result is reflected by the laser drop point. The test results are shown as Figure 17.



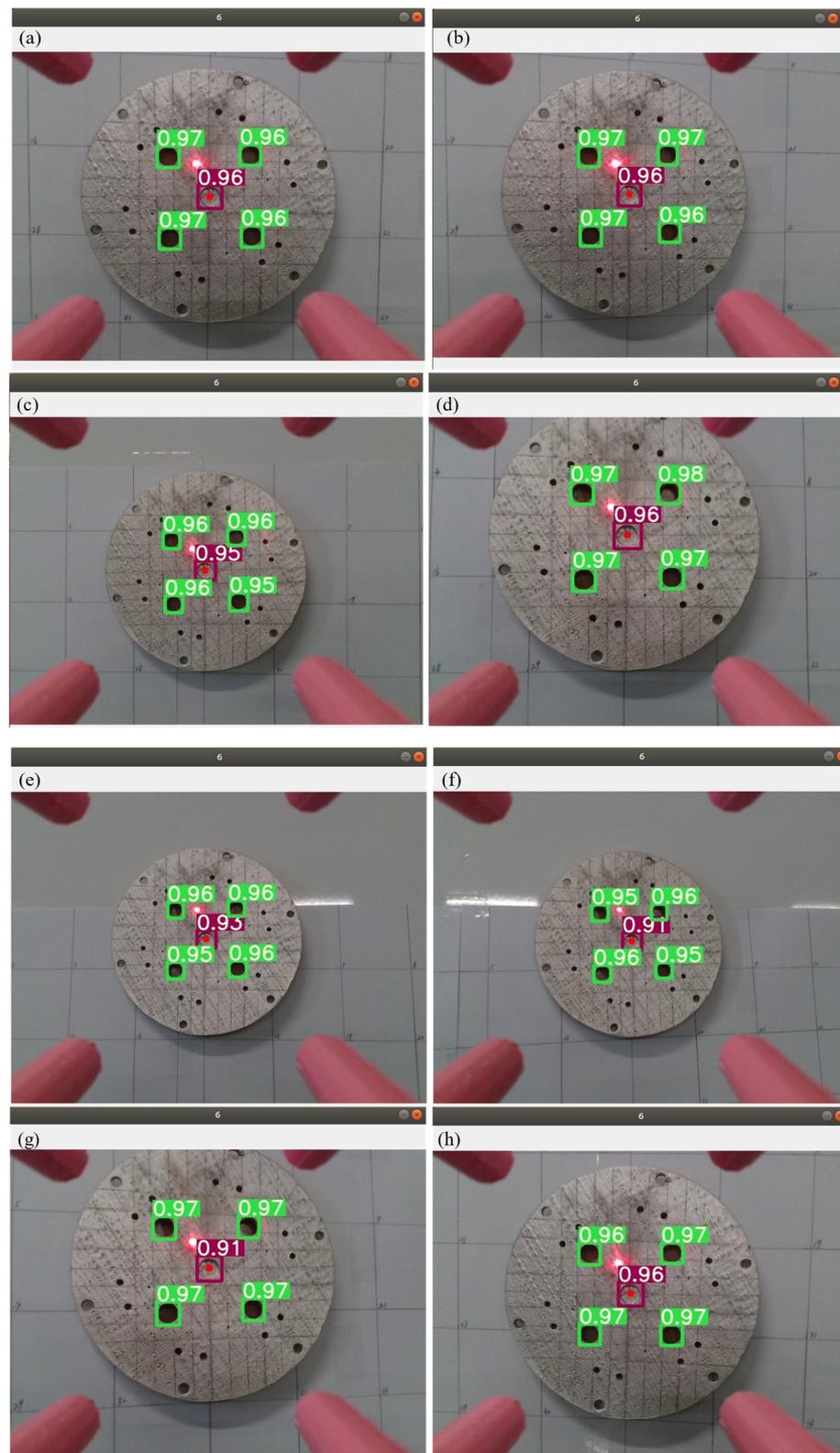
**Figure 17.** Disk tracking reality map: (a) grid lines in the neighborhood of point 42; (b) grid lines (each grid is 5 cm); (c) a porous disk with a diameter of 14 cm is moved on the magnetic plate with grid lines.

It can be seen from the results that the accuracy of target detection and tracking is relatively high, and the error between the moving laser drop point and the 42-point drop point is relatively small. It can be seen from Figure 18e,f that the laser drop point is tilted upward because the height at point 17 and point 19 is already at the limit of the mechanical arm's extension, so the end is in the upward attitude, and so, the laser drop point is tilted upward. As can be seen in Figure 18c,d, the laser drop point is lower because the camera and the fixed end of the mechanical arm have a certain weight, and the mechanical arm needs to overcome gravity to work, so the drop point is lower.

The paper also conducts tracking tests for multiple directions and multiple moving distances to further measure tracking errors. After model training, coordinate conversion, and communication between the upper and lower position machines, the location tracking experiment of the porous disk is carried out. The tracking accuracy of the porous disk (diameter 14 cm) was examined according to the horizontal displacement, vertical displacement, and a combined horizontal and vertical displacement, and data points were represented on the data laser point location map by small colored dots (0.15 cm in diameter), as follows:

- Red: initial point;
- Dark blue: the laser landing point after moving 5 cm;
- Green: laser landing point after moving 10 cm;
- Light blue: the laser landing point after moving 15 cm;
- Yellow: the laser drop point after moving 20 cm.

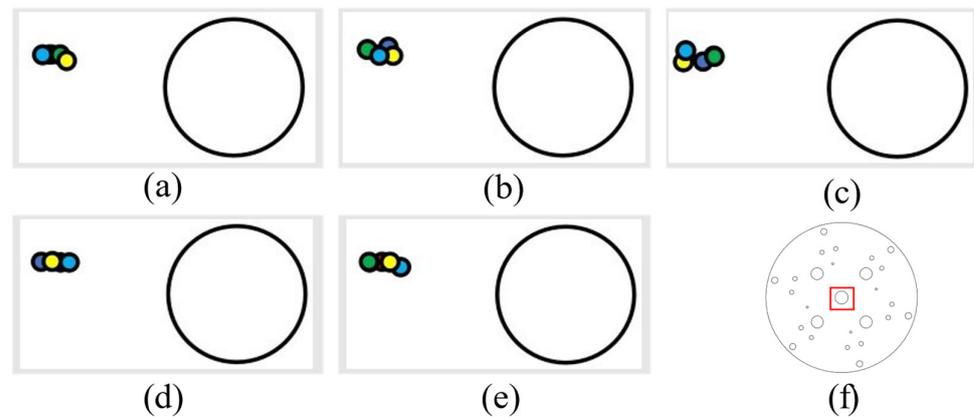
Tracking of horizontal displacements was conducted (5 cm to the left each time, with detection of error).



**Figure 18.** Dot 42 neighborhood trace: (a) at point 42, the laser landing point diagram; (b) move to 43; (c) move to 29; (d) move toward 30; (e) move toward 17; (f) move to 19; (g) move toward 31; (h) move to 41.

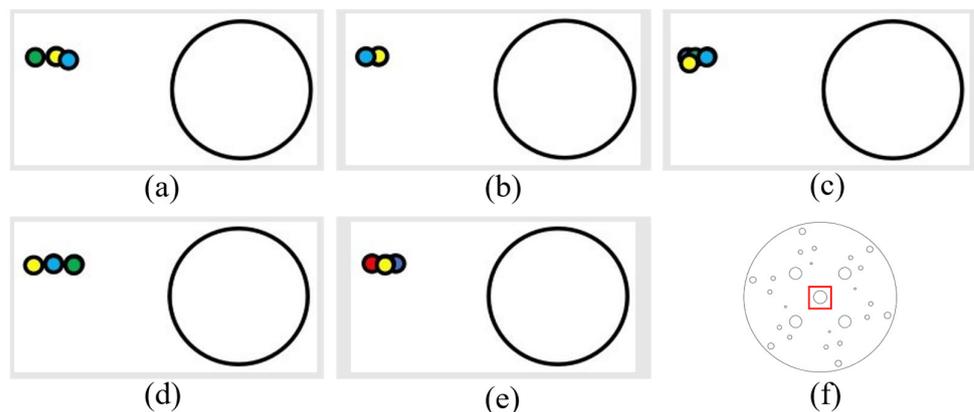
(1) The following figure shows five sets of data regarding leftward displacement and the position diagram of the laser drop point on the disk (showing an enlarged drop point

position map), where the large circle in Figure 19a,b is the circle within the red rectangular box of Figure 19f (with a diameter of 12.3 mm). All the following laser drop-point data position maps are graphed similarly.



**Figure 19.** Left-moving laser drop-point maps: (a–e) are five sets of left-moving drop-point data maps, and (f) is the whole-disk map.

(2) The following figure shows five sets of data regarding the rightward displacement and the position of the laser landing point. The figures are as described above (Figure 20).



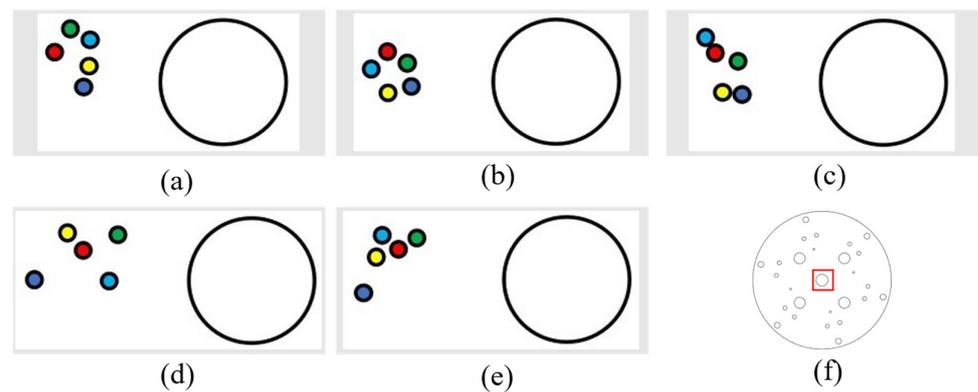
**Figure 20.** Right-moving laser drop-point maps: (a–e) are five sets of right-moving drop-point data maps, and (f) is the whole-disk map.

It can be determined, from the left and right movement data, that the laser landing points for horizontal tracking were relatively concentrated and that the error of horizontal tracking was less than 1.5 mm.

Next, the tracking of vertical displacements was also conducted (5 cm per vertical movement each time, with detection of error).

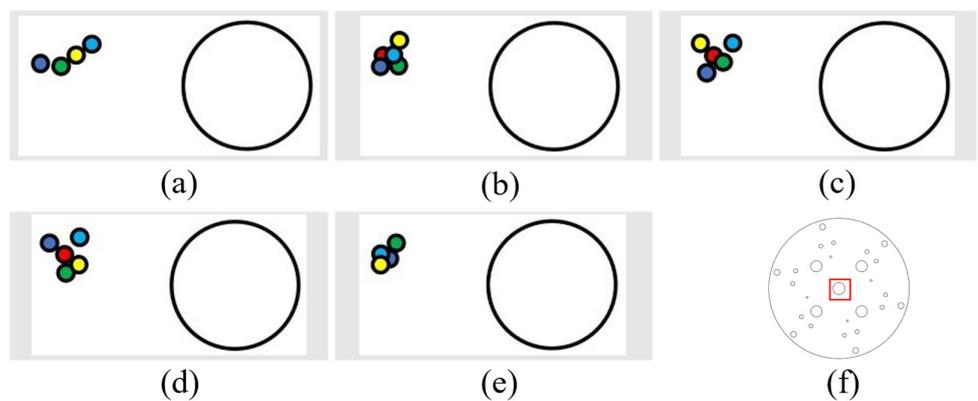
(1) The following figure shows five sets of data regarding upward displacement and the position of the laser drop point. The figures are as described above (Figure 21).

Through the upward movement of these groups of laser drop-point position maps, it can be seen that these drop points were relatively scattered; however, they were also located around the initial point, and the error was less than 5 mm. This is because, when the mechanical arm moves upward, the suspended mechanical arm has to overcome gravity to work due to its weight, and so, there is some shaking. For this reason, the error is larger than with the horizontal displacement.



**Figure 21.** Upward-moving laser drop-point maps: (a–e) are five sets of upward-moving drop-point data maps, and (f) is the whole-disk map.

(2) The following figure shows five sets of data regarding downward displacement and the position of the laser landing point. The figures are as described above (Figure 22).

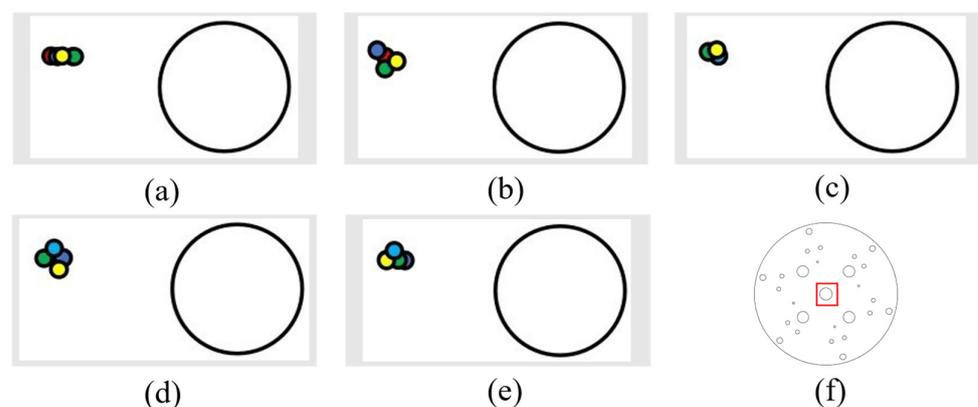


**Figure 22.** Downward-moving laser drop-point maps: (a–e) are five sets of downward-moving drop-point data maps, and (f) is the whole-disk map.

From the above sets of laser landing-point position diagrams, it can be seen that the landing points were relatively concentrated, and the error is less than 3 mm.

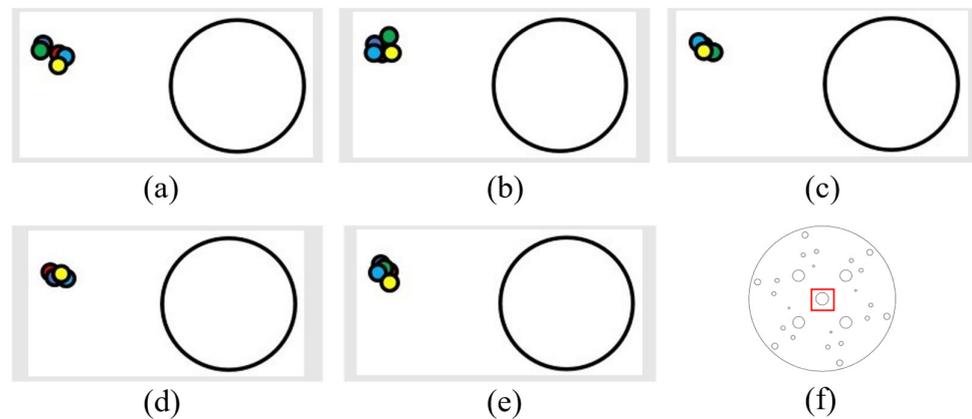
Next, combined horizontal and vertical motion tracking was conducted (5 cm horizontally and 5 cm vertically each time, with detection of error).

(1) The following figure shows five sets of data and laser drop-point location diagrams with both left and upward displacement. The figures are as described above (Figure 23).



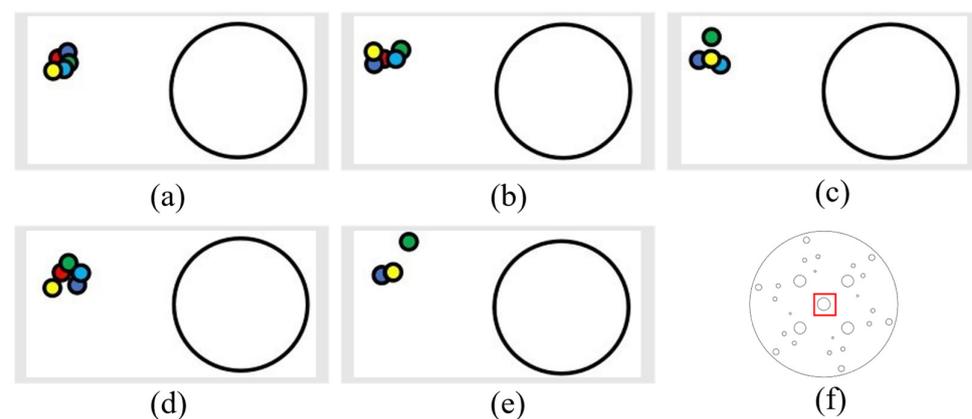
**Figure 23.** Left- and upward-moving laser drop-point maps: (a–e) are five sets of left- and upward-moving drop-point data maps, and (f) is the whole-disk map.

(2) The following figure shows five sets of data and laser drop-point location diagrams that are both displaced to the right and upward. The figures are as described above (Figure 24).



**Figure 24.** Right- and upward-moving laser drop-point maps: (a–e) are five sets of right- and upward-moving drop-point data maps, and (f) is the whole-disk map.

(3) The following figure shows five sets of data and laser drop-point location maps that are both left and downward. The figures are as described above (Figure 25).

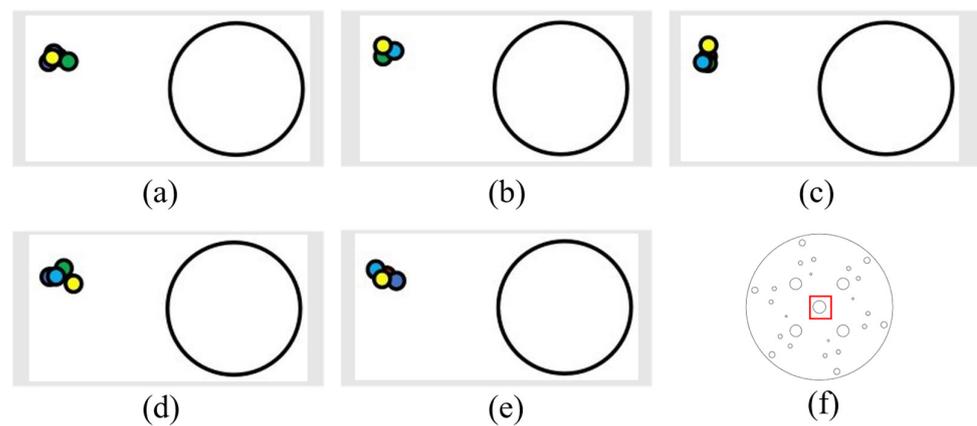


**Figure 25.** Left- and downward-moving laser drop-point maps: (a–e) are five sets of left- and downward-moving drop-point data maps, and (f) is the whole-disk map.

(4) The following figure shows five sets of data and laser drop-point location diagrams that are both displaced. The figures are as described above (Figure 26).

It can be seen from the above tracking laser drop location graphs combining the horizontal and vertical displacements that the laser drops were relatively concentrated. When this finding was combined with the analysis of the drop data in Appendix A, it could be concluded that, when the robotic arm was tracking the circular holes, the error was within 1.5 mm.

In summary, the multi-hole tracking and localization method proposed in this paper achieved accuracy within a 1.5 mm level of error, despite the limitations of the hardware used. It may be regarded, therefore, as a good foundation for automated assembly in various industries.



**Figure 26.** Right- and downward-moving laser drop-point maps: (a–e) are five sets of right- and downward-moving drop-point data maps, and (f) is the whole-disk map.

## 7. Conclusions

Aiming at the problem of large-scale automatic measurement and automatic assembly of the workpiece in a complex field environment, the system improves the anti-interference ability of the workpiece environment through the combination of machine vision and YOLOv5 deep learning. The robot arm is controlled by ROS topic communication in real time, and the porous disk is detected. The coordinates are transformed for real-time positioning and following operations, and the robot arm is controlled to track the hole position of the disc. This method has relatively high detection accuracy for small objects and small following error, which breaks the fixed and repetitive work of industrial robots, meets the needs of workpiece positioning on industrial production lines, provides a good prospect for the automation and intelligence of assembly lines, and also provides important ideas and insights for the assembly process of positioning small objects in industry.

However, due to the influence of hardware conditions and algorithms, the research described in this paper still has the following limitations, which need to be addressed in future studies:

1. The control accuracy of the servo was  $1^\circ$ . This was inadequate, as it required that some values be discarded when positioning. Some values should be discarded when positioning, and the steering gear with higher accuracy can be replaced by further research.
2. The training performance of the model was relatively large, and the training time was relatively long. Subsequently, it can be used for detection after training by optimizing algorithms or selecting cloud servers.

**Author Contributions:** Conceptualization, R.H. and Y.L.; methodology and simulation, R.H. and H.L.; validation, H.L.; formal analysis, R.H., Y.L. and H.L.; investigation, R.H.; resources, J.Y. and H.L.; data curation, H.L.; writing—original draft preparation, R.H.; writing—review and editing, J.Y. and Y.L.; visualization, R.H.; supervision, J.Y. and H.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Natural Science Foundation of Shanxi Province, China, grant number 202103021224187.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** No new data were created or analyzed in this study. Data sharing is not applicable to this article.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Appendix A

Table A1. Data with the point moving to the left.

Point Location Map Number	Class Number	Center Point Pixel	Deviation from the Initial Point Angle	Distance from the Initial Point
a	1	(313.0, 212.5)	0	0
Initial point center coordinates (313.0, 212.5)	2	(315.0, 211.5)	Due east	1 mm
	3	(312.0, 211.5)	Due west	0.5 mm
	4	(309.0, 210.5)	74° south by east	1.5 mm
	1	(314.5, 212.0)	60° north by east	1 mm
Initial point center coordinates (315.5, 210.5)	2	(316.5, 212.0)	60° north by west	1 mm
	3	(315.5, 210.5)	0	0
	4	(309.5, 210.5)	Due east	1 mm
	1	(313.0, 209.5)	67° south by west	1 mm
Initial point center coordinates (312.0, 210.0)	2	(312.5, 210.0)	0	0
	3	(313.0, 211.5)	56° north by west	1.5 mm
	4	(311.0, 210.0)	Due east	1.5 mm
	1	(318.5, 211.5)	Due west	1 mm
Initial point center coordinates (316.5, 212.0)	2	(318.5, 211.5)	Due west	1 mm
	3	(314.0, 210.5)	75° south by east	1.5 mm
	4	(315.5, 212.0)	Due east	0.5 mm
	1	(313.0, 217.0)	Due west	1 mm
Initial point center coordinates (312.5, 218.5)	2	(311.5, 218.5)	Due east	1.5 mm
	3	(310.5, 219.0)	Due east	1.5 mm
	4	(312.5, 218.5)	0	0

Table A2. Data with the point moving to the right.

Point Location Map Number	Class Number	Center Point Pixel	Deviation from the Initial Point Angle	Distance from the Initial Point
a	1	(311.0, 212.5)	0	0
Initial point center coordinates (310.0, 213.0)	2	(313.0, 212.5)	Due west	1.5 mm
	3	(308.5, 212.0)	76° south by east	1 mm
	4	(310.0, 213.0)	0	0
	1	(312.5, 213.5)	0	0
Initial point center coordinates (312.5, 214.0)	2	(314.5, 213.0)	Due west	1 mm
	3	(315.0, 213.0)	Due west	1 mm
	4	(314.5, 213.0)	Due west	1 mm
	1	(313.0, 209.5)	67° south by west	1 mm
Initial point center coordinates (314.0, 215.5)	2	(312.5, 210.0)	0	0
	3	(313.0, 211.5)	56° north by west	1.5 mm
	4	(311.0, 210.0)	Due east	1.5 mm
	1	(307.5, 213.5)	Due east	1.5 mm
Initial point center coordinates (310.5, 213.5)	2	(307.0, 213.0)	Due east	1.5 mm
	3	(310.0, 213.0)	0	0
	4	(312.5, 213.5)	Due west	1.5 mm
	1	(311.0, 210.5)	Due east	2 mm
Initial point center coordinates (314.0, 210.5)	2	(312.5, 210.5)	Due east	1 mm
	3	(312.0, 211.0)	Due east	1 mm
	4	(312.0, 211.0)	Due east	1 mm

Table A3. Data with an upward movement.

Point Location Map Number	Class Number	Center Point Pixel	Deviation from the Initial Point Angle	Distance from the Initial Point
a	1	(310.5, 206.0)	30° south by east	4 mm
Initial point center coordinates (314.0, 209.5)	2	(313.0, 212.0)	32° north by east	2.5 mm
	3	(309.0, 208.5)	73° north by east	3 mm
	4	(309.5, 207.5)	65° south by east	3 mm
	1	(314.5, 212.0)	60° north by east	1 mm
Initial point center coordinates (313.0, 211.0)	2	(316.5, 212.0)	60° north by west	1 mm
	3	(315.5, 210.5)	0	0
	4	(309.5, 210.5)	Due east	1 mm
	1	(308.0, 206.0)	31° south by east	5 mm
Initial point center coordinates (312.0, 209.5)	2	(309.5, 208.5)	64° south by east	2 mm
	3	(313.0, 210.0)	38° north by west	1.5 mm
	4	(309.0, 212.0)	11° south by west	3.5 mm
	1	(310.5, 206.5)	29° south by west	4.5 mm
Initial point center coordinates (314.0, 212.0)	2	(307.5, 210.0)	63° north by east	3 mm
	3	(308.0, 207.5)	41° south by east	3.5 mm
	4	(312.0, 211.0)	39° north by west	2.5 mm
	1	(308.5, 207.5)	28° south by west	4.5 mm
Initial point center coordinates (314.0, 210.0)	2	(310.5, 211.0)	60° north by east	2 mm
	3	(312.0, 212.0)	48° north by west	2 mm
	4	(313.5, 209.5)	71° south by west	2.5 mm

Table A4. Data with a downward movement.

Point Location Map Number	Class Number	Center Point Pixel	Deviation from the Initial Point Angle	Distance from the Initial Point
a	1	(313.0, 231.5)	80° south by west	3 mm
Initial point center coordinates (310.5, 233.5)	2	(312.0, 230.5)	60° south by west	1.5 mm
	3	(308.5, 234.0)	60° north by east	1.5 mm
	4	(310.0, 233.0)	0	0
	1	(314.0, 234.5)	19° south by west	1 mm
Initial point center coordinates (314.5, 234.0)	2	(313.0, 231.5)	54° south by east	1.5 mm
	3	(315.0, 231.0)	Due east	1 mm
	4	(315.0, 230.5)	50° north by east	2 mm
	1	(314.5, 230.0)	21° south by west	1.5 mm
Initial point center coordinates (313.0, 231.5)	2	(312.0, 231.5)	53° south by east	1 mm
	3	(312.0, 230.5)	57° north by east	2 mm
	4	(315.0, 232.5)	49° north by west	1.5 mm
	1	(311.5, 233.0)	47° north by west	1.5 mm
Initial point center coordinates (310.5, 232.5)	2	(310.0, 230.5)	3° south by east	1.5 mm
	3	(310.5, 236.0)	35° north by east	2 mm
	4	(310.5, 231.5)	56° south by east	1.5 mm
	1	(310.5, 235.5)	60° south by east	1 mm
Initial point center coordinates (313.0, 238.0)	2	(312.0, 234.0)	56° north by east	1.5 mm
	3	(313.5, 237.0)	0	0
	4	(312.5, 235.0)	Due south	1 mm

Table A5. Data with both left and right movement.

Point Location Map Number	Class Number	Center Point Pixel	Deviation from the Initial Point Angle	Distance from the Initial Point
a	1	(316.5, 215.5)	Due east	2 mm
Initial point center coordinates (320.0, 216.5)	2	(317.0, 215.5)	Due east	2 mm
	3	(318.5, 216.0)	Due east	0.5 mm
	4	(317.5, 215.0)	Due east	1 mm
	1	(313.0, 208.5)	56° north by west	1 mm
Initial point center coordinates (312.0, 207.5)	2	(312.0, 206.0)	Due south	1 mm
	3	(311.5, 207.0)	63° south by east	1 mm
	4	(311.5, 207.0)	63° south by east	1 mm
	1	(312.0, 215.0)	65° north by west	1 mm
Initial point center coordinates (310.5, 214.0)	2	(312.0, 215.0)	65° north by west	1 mm
	3	(310.5, 213.5)	0	0
	4	(311.5, 214.0)	21° north by west	0.5 mm
	1	(315.0, 215.0)	0	0
Initial point center coordinates (315.0, 215.0)	2	(317.0, 215.0)	Due west	1.5 mm
	3	(316.5, 217.5)	40° north by west	1 mm
	4	(316.0, 214.0)	26° south by west	1 mm
	1	(310.0, 212.0)	Due east	1.5 mm
Initial point center coordinates (312.5, 212.0)	2	(311.5, 211.5)	Due east	1 mm
	3	(311.0, 213.0)	42° north by east	1 mm
	4	(312.0, 212.5)	0	0

Table A6. Data with both right and upward movement.

Point Location Map Number	Class Number	Center Point Pixel	Deviation from the Initial Point Angle	Distance from the Initial Point
a	1	(310.0, 219.0)	52° north by west	1.5 mm
Initial point center coordinates (307.5, 216.0)	2	(305.5, 215.0)	75° north by west	1.5 mm
	3	(311.0, 217.0)	52° south by east	0.5 mm
	4	(308.0, 218.0)	Due north	1 mm
	1	(312.0, 217.0)	30° north by west	1 mm
Initial point center coordinates (311.5, 216.0)	2	(310.5, 218.0)	22° north by east	1.5 mm
	3	(312.0, 215.5)	Due west	0.5 mm
	4	(310.5, 215.0)	Due east	0.5 mm
	1	(311.5, 217.5)	50° north by west	1 mm
Initial point center coordinates (309.5, 215.5)	2	(310.0, 215.0)	0	0
	3	(312.0, 218.0)	50° north by west	0.5 mm
	4	(312.0, 216.0)	76° north by west	1 mm
	1	(315.5, 216.0)	40° south by east	0.5 mm
Initial point center coordinates (316.0, 217.0)	2	(313.5, 216.5)	74° south by east	1 mm
	3	(312.5, 215.5)	70° south by east	1.5 mm
	4	(314.0, 214.0)	74° south by east	1 mm
	1	(314.5, 215.0)	44° north by west	1 mm
Initial point center coordinates (313.5, 213.0)	2	(314.0, 214.0)	44° north by west	0.5 mm
	3	(315.0, 213.0)	Due west	1 mm
	4	(313.5, 215.0)	Due north	1 mm

Table A7. Data with both left and downward movement.

Point Location Map Number	Class Number	Center Point Pixel	Deviation from the Initial Point Angle	Distance from the Initial Point
a	1	(318.0, 240.0)	58° north by east	1 mm
Initial point center coordinates (319.5, 239.0)	2	(318.5, 238.0)	65° south by east	1 mm
	3	(318.0, 238.5)	21° south by east	1 mm
	4	(321.0, 238.5)	27° south by west	1 mm
	1	(315.0, 235.0)	62° south by west	1 mm
Initial point center coordinates (313.0, 236.0)	2	(310.5, 237.5)	62° north by east	1.5 mm
	3	(311.0, 235.5)	Due east	1 mm
Initial point center coordinates (316.5, 235.0)	4	(315.0, 237.0)	53° north by west	1 mm
	1	(317.0, 235.0)	Due west	1 mm
	2	(316.5, 238.0)	Due north	2 mm
	3	(316.0, 234.0)	55° south by east	1 mm
Initial point center coordinates (315.0, 236.0)	4	(316.5, 235.0)	0	0
	1	(314.0, 234.5)	51° south by east	1.5 mm
	2	(314.5, 237.5)	33° north by east	1 mm
	3	(313.0, 236.0)	Due east	1.5 mm
Initial point center coordinates (317.0, 233.0)	4	(316.5, 234.5)	33° south by west	1.5 mm
	1	(317.0, 233.0)	0	0
	2	(314.5, 236.5)	40° north by east	3.5 mm
	3	(316.0, 234.0)	69° north by east	1 mm
	4	(316.0, 234.0)	69° north by east	1 mm

Table A8. Data with both right and downward movement.

Point Location Map Number	Class Number	Center Point Pixel	Deviation from the Initial Point Angle	Distance from the Initial Point
a	1	(315.5, 236.0)	66° south by west	1 mm
Initial point center coordinates (314.0, 238.0)	2	(313.0, 238.5)	72° south by east	1 mm
	3	(314.5, 239.5)	41° north by west	0.5 mm
	4	(315.0, 237.0)	Due west	0.5 mm
	1	(313.5, 234.0)	Due north	1 mm
Initial point center coordinates (313.5, 231.0)	2	(314.0, 231.5)	0	0
	3	(311.5, 233.5)	60° north by west	1 mm
	4	(313.5, 234.5)	Due north	1 mm
	1	(311.5, 237.5)	40° south by west	0.5 mm
Initial point center coordinates (310.0, 238.5)	2	(310.5, 236.5)	19° south by west	0.5 mm
	3	(311.0, 237.0)	44° south by west	0.5 mm
	4	(310.5, 241.0)	Due north	1 mm
	1	(313.0, 234.5)	Due west	0.5 mm
Initial point center coordinates (312.5, 235.0)	2	(311.5, 236.5)	40° north by east	1 mm
	3	(312.0, 235.0)	0	0
	4	(310.0, 237.0)	70° south by east	1.5 mm
	1	(310.5, 233.5)	60° south by east	1 mm
Initial point center coordinates (312.5, 235.5)	2	(315.0, 234.5)	64° north by west	1 mm
	3	(314.5, 235.0)	64° north by west	1 mm
	4	(313.5, 233.0)	54° south by west	0.5 mm

## References

1. Andhare, P.; Rawat, S. Pick and Place Industrial Robot Controller with Computer Vision. In Proceedings of the 2016 International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 12–13 August 2016; pp. 1–4.
2. Shahzad, A.; Gao, X.; Yasin, A.; Javed, K.; Anwar, S.M. A Vision-Based Path Planning and Object Tracking Framework for 6-DOF Robotic Manipulator. *IEEE Access* **2020**, *8*, 203158–203167. [\[CrossRef\]](#)
3. Ding, Y.; Hua, L.; Li, S. Research on computer vision enhancement in intelligent robot based on machine learning and deep learning. *Neural Comput. Appl.* **2022**, *34*, 2623–2635. [\[CrossRef\]](#)
4. Yang, Y.; Liu, Y.H.; Peng, Y.; Sun, Y.S.; Zhang, R.F. Design of Object Recognition and Tracking Algorithm and System Based on Machine Vision. *Transducer Microsyst. Technol.* **2020**, *39*, 92–95+98.

5. Lin, L.G.; Liu, Y.; Li, J.; Liu, X. Motion Tracking Model Of 6-DOF Manipulator Based on Trajectory Measurement and Human-machine Mapping. *J. Electron. Meas. Instrum.* **2023**, *37*, 102–110.
6. Jin, X.L. Xinsong Industrial Robot Helps Intelligent Upgrading of China’s Shipbuilding Industry. *Liaoning Daily News*, 29 January 2019.
7. Han, S. Kinematics Analysis and Research of Manipulator Based on Machine. Master’s Thesis, Beijing Jiaotong University, Beijing, China, 2021.
8. Lu, Y.L. Research and Development of Flexible Processing System Based on KUKA Industrial Robot. *Value Eng.* **2019**, *38*, 87–91.
9. Xiao, X.; Huang, Z.; Rube, M.A.; Melzer, A. Investigation of Active Tracking for Robotic Arm Assisted Magnetic Resonance Guided Focused Ultrasound Ablation. *Int. J. Med. Robot. Comput. Assist. Surg.* **2017**, *13*, e1768. [[CrossRef](#)] [[PubMed](#)]
10. Guo, J.; Luo, H.; Zhang, D.D. Research on a High Precision Screw Position Detection Method Based on Machine Vision. *Navig. Control* **2016**, *15*, 106–112.
11. Li, T.; Liu, Y. Study on Circle Locating Technology Based on Machine Vision. *Comput. Eng. Appl.* **2012**, *48*, 153–156.
12. Luo, H.; Cui, Y.F. Design and Implementation of Workpiece Recognition and Positioning System Based on Machine Vision. *Modul. Mach. Tool Autom. Manuf. Tech.* **2021**, *1*, 101–105+109.
13. Baduge, S.K.; Thilakarathna, S.; Perera, J.S.; Arashpour, M.; Sharafi, P.; Teodosio, B.; Shringi, A.; Mendis, P. Artificial intelligence and smart vision for building and construction 4.0: Machine and deep learning methods and applications. *Autom. Constr.* **2022**, *141*, 104440. [[CrossRef](#)]
14. Bogue, R. The role of machine learning in robotics. *Ind. Robot.* **2023**, *50*, 197–202. [[CrossRef](#)]
15. Zhao, S.L. Train Hook Removal Robot Based on Visual Target Tracking Algorithm. *J. Pop. Electr.* **2023**, *38*, 62–63.
16. Zhao, Z.Y. Research on Dynamic Target Detection and Tracking Algorithm of Workpiece Sorting Robotic Arm. Master’s Thesis, Lanzhou Jiaotong University, Lanzhou, China, 2022.
17. Maschler, B.; Weyrich, M. Deep Transfer Learning for Industrial Automation: A Review and Discussion of New Techniques for Data-Driven Machine Learning. *IEEE Ind. Electron.* **2020**, *15*, 65–75. [[CrossRef](#)]
18. Ohwovoriole, M.S. An Extension of Screw Theory. *J. Mech. Des.* **1981**, *103*, 725–735. [[CrossRef](#)]
19. Yang, J. Research on Multi-hole Positioning System of Panel Light Based on Machine Vision. Master’s Thesis, Anhui University of Technology, Ma’anshan, China, 2019.
20. Hadi, R.H.; Hady, H.N.; Hasan, A.M.; Al-Jodah, A.; Humaidi, A.J. Improved Fault Classification for Predictive Maintenance in Industrial IoT Based on AutoML: A Case Study of Ball-Bearing Faults. *Processes* **2023**, *11*, 1507. [[CrossRef](#)]
21. Cheng, Y.L.; Zhang, H.W.; Wang, X.H.; Guo, Z. Research on Riveting Hole Positioning Technology of Tramcar Truss Based on YOLO-V5. *Mod. Manuf. Eng.* **2022**, 115–121.
22. Malta, A.; Mendes, M.; Farinha, T. Augmented Reality Mainte Assistant Using YOLO-V5. *Appl. Sci.* **2021**, *11*, 4758. [[CrossRef](#)]
23. Shao, Y.H.; Zhang, D.; Chu, H.Y.; Zhang, X.; Rao, Y. A Review of YOLO Object Detection Based on Deep Learning. *J. Electron. Inf. Technol.* **2022**, *44*, 3697–3708.
24. Wu, P.K.; Leng, X.H.; Wu, J.L.; Cheng, J.Z. A joint calibration method for camera and lidar based on key points. *J. Agric. Equip. Veh. Eng.* **2023**, *61*, 58–62.
25. Shi, Y.Q. Research on Application of Machine Vision Technology in Mold Inspection and Positioning. Master’s Thesis, University of Electronic Science and Technology of China, Chengdu, China, 2020.
26. Qiu, S.; Yang, C.H.; Wu, L.; Wang, K.-C.; Pan, J.-Z. Machine-vision-based Spindle Positioning System of Grinding-wheel-saw Automatic Replacement System. *Sens. Mater.* **2022**, *34*, 789–801. [[CrossRef](#)]
27. Wu, G.Y.; Zhang, W.J.; Yang, B.J. Research on Accurate Positioning Method in Robot Visual Assembly System. *Mach. Electron.* **2020**, *38*, 75–80.

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