Journal of Computers Vol. 35 No. 1, February 2024, pp. 207-215
doi: 10.53106/199115992024023501016

Optimization Method for Robot Moving Object Recognition and Grasping Strategy Based on Binocular Vision

Xiao-Yang Zhang[1,2], Rui Fan[1,2*], Wei-Min Liu[1,2], Jian-Fang Xue[1], Qing-Chuan Liu[1,2]

[1] Hebei Institute of Mechanical and Electrical Technology,
Xingtai City 054000, Hebei Province, China
{xiaoyang86863, fanrui98792, weimin78687, jianfang68789,
qingchuan3978}@126.com

[2] Xingtai Electromechanical Equipment Intelligent Perception and Intelligent Control Technology Innovation Center,
Xingtai City 054000, Hebei Province, China

Received 15 December 2023; Revised 15 January 2024; Accepted 31 January 2024

Abstract. This article proposes a more accurate grasping strategy for the recognition and grasping of moving targets based on binocular vision cameras. Firstly, the front and back scene separation algorithm is used to identify the moving target grabbing object in the production line. Then, by setting an appropriate threshold, the SiamMask target tracking algorithm is improved to achieve dynamic target tracking. Finally, the conveyor belt speed is detected and the real-time position of the object is obtained. Then, the Cartesian strategy is used to achieve path planning and optimization methods for the robotic arm during movement. Through experimental simulation, the effectiveness and stability of the proposed method in this paper have been demonstrated.

Keywords: dynamic crawling, dynamic tracking, binocular vision, collaborative robots

1 Introduction

Industrial robots play a crucial role in today's intelligent manufacturing field, and robot grasping is an extremely common link in the manufacturing industry. In actual production, automated production lines run continuously. If the robot grabs, the production line needs to stop, and the goal of the robot grabbing stationary is conventional operation. However, the production line stopping will inevitably affect production efficiency, Therefore, achieving grasping tasks under the premise of normal operation of the production line is an effective means to improve production efficiency.

There are two main methods for industrial robots to grasp moving targets: one is delay compensation grasping, which uses a visual sensor to obtain the target position in a single attempt and combines distance compensation with the time difference between taking photos and grasping to achieve the grasping task. It is susceptible to the influence of conveyor line vibration and target posture changes, resulting in a high failure rate of grasping; Another method is tracking and grasping, which obtains the target position in real-time through visual sensors and guides the robot to track and grasp. It can effectively overcome the problems of conveyor line vibration and target attitude changes, and has a high success rate in grasping.

Cameras, similar to the human eye, can obtain rich and colorful environmental information such as texture and color, and the cost of cameras is relatively low. Therefore, visual recognition has great research prospects and is a direction for future research. Binocular vision positioning is the most widely used method of positioning and recognition. Binocular vision positioning refers to obtaining the same feature from different perspectives and using triangulation algorithms to obtain its disparity, thereby obtaining depth and position information of the captured target.

This article is based on binocular vision and focuses on the study of continuously moving grasping targets in automated production lines. It proposes precise recognition and positioning methods, and then develops tracking strategies for moving targets. Finally, path planning is carried out to achieve accurate recognition and grasping of target objects. Therefore, the work of this article is as follows:

* Corresponding Author

1) Using the front and back scene separation algorithm to identify moving objects in the production line, and improving the recognition method to accurately identify dynamic moving targets.

2) Select the SiamMask target tracking algorithm to solve the problem of being susceptible to background or other noise interference, improve the robustness and real-time performance of tracking, and lay the foundation for subsequent robot dynamic target grasping.

3) Research the detection of conveyor belt speed and obtaining real-time position of objects, then study the path planning and optimization methods of the robotic arm during the motion process, and finally achieve the dynamic grasping of moving targets by the robotic arm with the assistance of the visual system.

The composition and structure of this article are as follows: Chapter 2 mainly introduces the relevant research work, mainly in terms of dynamic recognition algorithms. Chapter 3 discusses the tracking strategy for dynamic targets. Chapter 4 introduces the path planning method in the dynamic grasping process, and Chapter 5 is the experimental process to prove the effectiveness of the proposed method. Chapter 6 is the conclusion section, summarizing the results and describing the shortcomings.

2 Reated Work

Scholars and staff have done a lot of work on the recognition and capture of dynamic targets, and various methods have been used to improve recognition accuracy. Peng Wang used a robot sorting scheme during the coal gangue sorting process, and attempted to solve the problems of delay, shaking, and impact that occurred during the tracking and sorting of coal gangue by the robot. He used PID control algorithm to achieve dynamic grasping of coal gangue by the robot [1]. Min Wang proposed a multi-objective motion optimization strategy based on an improved particle swarm optimization algorithm for a high-speed parallel food sorting robot, which achieves the establishment of the shortest path model for dynamic targets, coordinated grasping sequence and sorting trajectory in food processing and sorting. The final conclusion is that, upon verification, the success rate of grasping increases from 96.8% to 100% at a conveying speed of 100mm/s [2]. Hongwei Ma, in response to issues such as inaccurate positioning of coal gangue due to conveyor belt slipping and swinging left and right, failure of mechanical arm end grasping, and load impact, first uses image algorithms to dynamically identify the posture of coal gangue, then establishes inverse kinematic equations, and then uses PID closed-loop control to achieve precise grasping. This strategy achieves an average speed deviation of around 1 mm/s for robot following and grasping [3]. Zhaoquan Wang proposed a dynamic grabbing method for traffic cones based on Kalman filtering and position servo, aiming at the grabbing problem of robotic arms in the automatic grabbing and releasing system of traffic cones. The relative motion model of the cones was established, and the Kalman filtering algorithm was used to predict the position of the cones. A grabbing method with speed compensation was proposed. The experimental results showed that this method can accurately and smoothly complete the grabbing [4]. Yizhong Ge proposed a monocular visual servo system based on the A network model. After obtaining the pose of the target from the monocular camera, the neural network B was used to learn and grasp the inverse kinematics of the robotic arm. Finally, experimental verification was completed using the C robot platform [5]. Qin Wan mainly studied how to accurately and stably follow targets in complex scenes. He used an improved YOLOX mobile robot target following method, added a Kalman filter to the algorithm, and finally completed mobile robot target following experiments in real scenes. The experimental results all verified that the proposed method has good robustness and real-time performance [6].

3 Dynamic Target Detection and Tracking Strategy

When the target grabbing object moves on the assembly line, the dynamic image captured by binocular vision is first preprocessed to filter out other irrelevant backgrounds and minimize the interference of other factors. This article uses a visual background extraction algorithm with strong adaptability to the environment and high processing efficiency for foreground and background separation, and improves the incomplete phenomenon of foreground extraction that occurs.

3.1 Background Extraction of Binocular Vision Images

This article uses the Visual Background Extraction Algorithm (ViBe) [7], which is achieved by randomly selecting neighboring pixels. This processing method can effectively reduce computational complexity, improve processing speed, and simulate the random changes of pixels when background pixels fluctuate.

1) Initialize Background Model

In the initialization of the background model, only the first frame is needed, and some pixels within a certain range of each pixel can be randomly selected to fill the background model.

2) Establishing a background model

A background model can be established by using N pixel values near the pixel point, as follows:

$$P(x)=\left\{P_1(x),\cdots,P_k(x),\cdots,P_N(x)\right\}. \tag{1}$$

$P(x)$ is the background model of pixel x, defining a two-dimensional Euclidean space $O_R(I_t(x))$ with $I_t(x)$ as the center and R as the radius. $I_t(x)$ represents the pixel value of pixel x at time t, and the specific meaning of pixel x at time t is represented as:

$$H(x)=\begin{cases}1 & \left\{O_R\left(I_t(x)\right)\cap\left\{p_1,p_2,\cdots,p_n\right\}\le t_{\min}\right\}\\ 0 & else\end{cases}. \tag{2}$$

In the formula, H represents the meaning of the pixel point, $H = 1$ represents the foreground, $H = 0$ represents the background, and $t_{\min}$ represents the judgment threshold. If there are more than $t_{\min}$ sampling data in $P(x)$ within $O_R(I_t(x))$, then the point is determined as the background point, otherwise it is the front attraction. Set N =20, $t_{\min} = 2$.

3) Background model update

When updating the background model, only the pixels judged as the background are considered, and the previous attractions are not updated into the model. A pixel judged as the background is used to randomly replace a sample in the background model B. If a pixel is judged as the previous attraction, it is not considered during the update process, and the update probability is expressed as:

$$P(t_0,t_1)=\left(\frac{n-1}{n}\right)^{(t_1-t_0)}. \tag{3}$$

As time continues to lengthen, the probability of each pixel not being updated decreases. When updating the pixel determined as the background into the model, this background point is used to replace a sample value in the background model of a certain pixel within a certain range of the point, so that the neighborhood near the pixel is also propagated during the background model update process.

4) Algorithm improvement

Based on the above, calculate the area of non parent contours and set a threshold. If the area of the speckled area in the current scene falls below this threshold, it will be removed. Set another threshold, and when the small holes in the current scene target are below this threshold, fill them up [8]. This can fill the background area with pixels that were detected incorrectly, reducing small noise in the background and making the background cleaner. The threshold for removing foreground spots is set to 20 pixels, while the threshold for filling foreground void areas is set to 40 pixels.

3.2 Target Extraction and Tracking

Determine the tracking target in the initial frame, detect each target image, extract target features, match the tracking target of the previous frame image, and determine the target motion state. Predict the target position in

the next frame of the image based on the target motion state to achieve target tracking. This article uses Model SiamMask [9] as the basic algorithm for target tracking, adds a self-made target tracking dataset to the algorithm, and trains the algorithm model. The model structure is shown in Fig. 1.

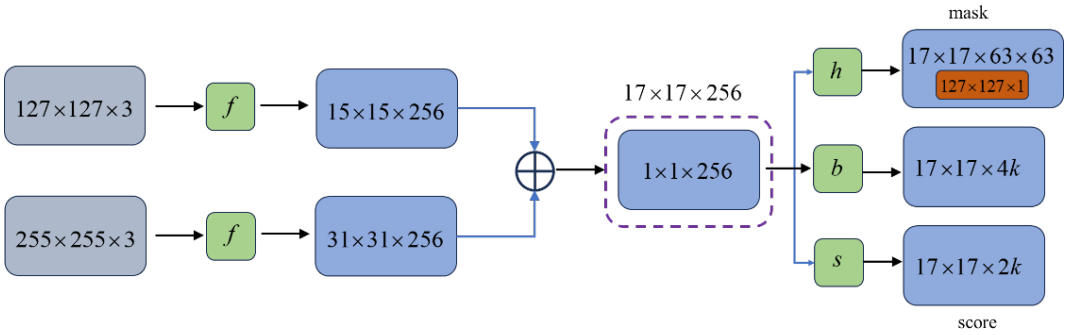

Fig. 1. Algorithm model structure

Adjust the input size of the template image to 127 × 127, adjust the input size of the search image to 255 × 255, use RGB three channels to process all images. The dataset contains a total of 100 video sequences, divided into a training set and a testing set. The training set is used to train the network model, while the testing set is used to evaluate the final network model and detect model performance. The network model is trained with a total of 100 Epoch, batch_size set to 2, the initial learning rate to 0.01, and use the Adam adaptive moment estimation optimizer to adjust the learning rate in real-time, with momentum set to 0.9 and attenuation set to 0.0005. Using the segmentation loss function to train the model, corresponding to the output of each branch, the formula is:

$$L_{mask}(\theta,\phi)=\sum_{n}\left(\frac{1+y_n}{2wh}\sum_{ij}\log\left(1+e^{-c_n^{ij}m_n^{ij}}\right)\right). \tag{4}$$

Using binary logistic regression to calculate segmentation loss, label each *RoW* with a standard binary label $y_n \in \{\pm 1\}$, and when $y_n = 1$ is used, it indicates that the sample is a positive sample. The segmentation mask loss function reflects the accuracy of the mask estimation model. The smaller the loss value, the more accurate it is, and vice versa, the less accurate it is. For each candidate sub window of the search image, the network model outputs a score indicating the probability that the window belongs to positive or negative samples. Therefore, the model transforms the tracking problem into a binary classification problem. At the same time, the value of the overall score graph is used as the classification loss function of the model, which is the average of all point classification loss functions. The loss function is represented as:

$$L_{score}=\frac{1}{|D|}\sum_{\mu\in D}l\left(y[\mu],v[\mu]\right). \tag{5}$$

In the equation, $\mu \in D$ represents the position of the point in the score map, D represents the size of the thermodynamic map, and the final loss function is represented as:

$$L=\alpha_1 L_{\max}+\alpha_2 L_{score},\alpha_1=32,\alpha_2=1. \tag{6}$$

The program flowchart of the entire algorithm is shown in Fig. 2.

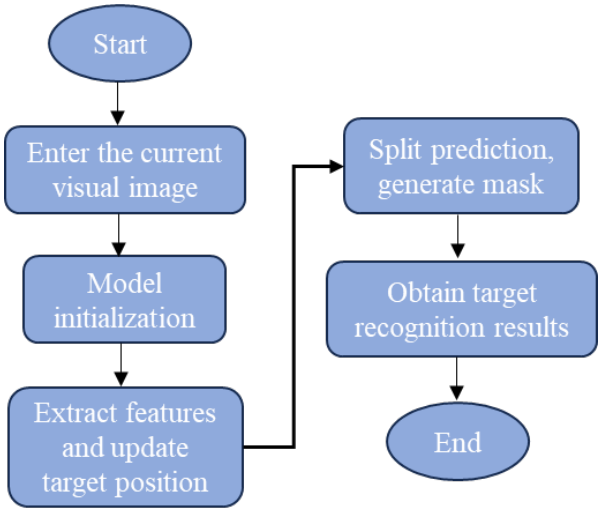

Fig. 2. Algorithm flow chart

4 Target Dynamic Grasping and Path Optimization

Researching the detection of conveyor belt speed and obtaining real-time positions of objects, then studying the path planning and optimization of the robotic arm during movement, and finally dynamically grasping moving targets with the assistance of the visual system.

4.1 Conveyor Speed Detection

Due to the fact that objects do not deviate relative to the belt when running on the conveyor belt, the key is to determine the speed of the conveyor belt, which is a prerequisite for obtaining accurate object movement speed. The tracking line speed is measured using Keinz's CL-E100 incremental encoder. During a very short time period Δt , the conveyor belt speed is:

$$v = \frac{2\pi \times \Delta n \times r}{N \times \Delta t}. \tag{7}$$

In the formula, Δn represents the pulse change value of the encoder in Δt , r represents the radius of the DC motor drive shaft driving the drive belt, and N is the number of pulses per revolution of the encoder.

4.2 Moving Target Position Tracking

Assuming that the system time when the object is captured is t_0 , and the position of the object in the camera coordinate system is (x_0, y_0, z_0), it is assumed that the robotic arm grasps the object at time t_1 . Since the motion direction of the conveyor belt is parallel to x in the camera coordinate system, the position of the object at time t_1 is:

$$\begin{cases} x_1 = x_0 + v(t_1 - t_0) \\ y_1 = y_0 \\ z_1 = z_0 \end{cases}. \tag{8}$$

In the above equation, is the conveyor belt speed, which is the position coordinate of the object on the conveyor belt at that time. After receiving the precise position of the target object sent by the visual system at the control end of the robotic arm, it will control the robotic arm to reach the grasping point at the specified time, and then open the gripper to grasp the object.

This article selects the $\cap$-type acceleration and deceleration control algorithm. The method achieves the acceleration and deceleration process of the robotic arm by segmented control of its motion speed. The speed appearance is $\cap$-shaped, the acceleration appearance is trapezoidal, the magnitude of the acceleration is constant, and the appearance fluctuates in the form of a horizontal line. The curve is shown in Fig. 3.

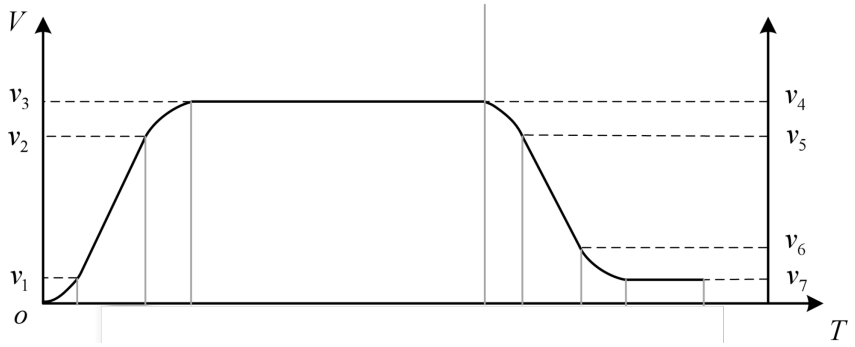

Fig. 3. Velocity fluctuation curve

By integrating the acceleration rate and combining it with the information in the figure above, the expression of the acceleration over time is obtained:

$$a(t)=\begin{cases} A_1\cdot t & 0\le t\le t_1 \\ a_{m1} & t_1\le t\le t_2 \\ a_{m1}+A_2(t-t_2) & t_2\le t\le t_3 \\ 0 & t_3\le t\le t_4 \\ a_{m2}-A_3(t-t_4) & t_4\le t\le t_5 \\ a_{m2} & t_{5\le}\le t\le t_6 \\ A_4(t-t_6)+a_{m_2} & t_6\le t\le t_7 \end{cases}. \tag{9}$$

By integrating the acceleration rate, the expression for the variation of speed over time is obtained:

$$\begin{cases} v_s+\frac{1}{2}A_1t^2 & 0\le t\le t_1 \\ v_1+a_{m1}(t-t_1) & t_1\le t\le t_2 \\ v_2+a_{m1}(t-t_2)-\frac{1}{2}A_2(t-t_2)^2 & t_2\le t\le t_3 \\ v_3 & t_3\le t\le t_4 \\ v_4-\frac{1}{2}A_3(t-t_4)^2 & \\ v_5-a_{m2}(t-t_5) & \\ v_6-a_{m2}(t-t_6)+\frac{1}{2}A_4(t-t_6)^2 & t_6\le t\le t_7 \end{cases}. \tag{10}$$

After receiving the coordinate position, the robotic arm controller controls the displacement of the robotic arm to the grasping point and plans the path to fully grasp the target object, enabling the robotic arm to stably and accurately grasp the target and proceed to the next step of operation. Considering that the end effector is an electric gripper, the center point of the gripper should be on the same vertical line as the center point of the object surface in the horizontal direction. The schematic diagram of the grabbing process is shown in Fig. 4.

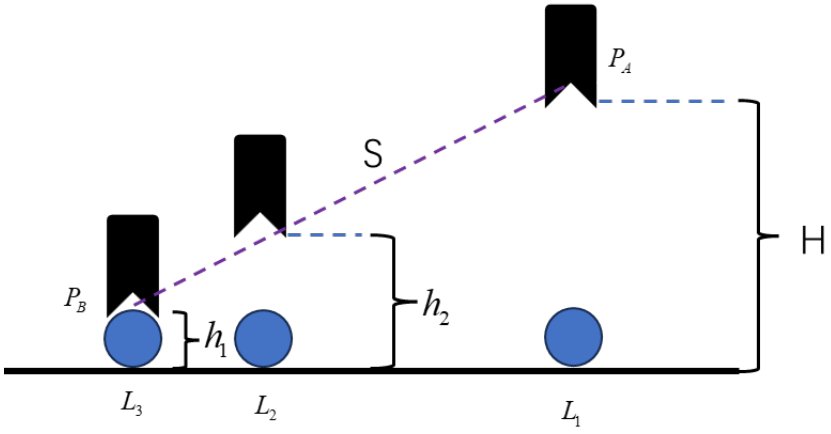

Fig. 4. Grab schematic diagram

This article uses the Cartesian path planning method, which only requires the accuracy of target position positioning and does not pay attention to the motion path of the end effector to the target position [10]. Assuming the starting point $P_A(x_A, x_B, x_C)$ and the ending point $P_B(x_B, y_B, z_B)$, the path straight line equation can be expressed as:

$$S = (y_B - y_A) \times (x - x_B) / (x_B - x_A) + y_B . \tag{11}$$

Using the number of interpolation points to complete path planning control, insert n equidistant calculation points in path segment $\overline{P_A P_B}$, and the interpolation step length and each interpolation point on the x-axis and y-axis are represented as follows:

$$\Delta x = (x_B - x_A) / (n+1) . \tag{12}$$

$$\Delta y = (y_B - y_A) / (n+1) . \tag{13}$$

$$\begin{cases} x_i = x_A + i \cdot \Delta x \\ y_i = y_A + i \cdot \Delta y \end{cases} . \tag{14}$$

If the interpolation distance on path segment $\overline{P_A P_B}$ is d, n interpolation transition points can be obtained. The interpolation step on the x-axis is $\Delta x = (x_B - x_A)/(n' + 1)$, and the interpolation step on the y axis is $\Delta y = (y_B - y_A)/(n' + 1)$.

5 Dynamic Grasping Experimental Results and Analysis

This article uses a collaborative robot as an experimental platform. After the conveyor belt is activated, the target object moves forward at a nearly uniform speed with the conveyor belt. The industrial camera faces the target

object and collects RGB data streams from the workspace. The robot tracks and grabs the dynamic target after receiving the visual recognition of the object's pose.

The total length of the conveyor belt is 0.9m, with an effective working range of 0.75m. It is driven by a DC speed control motor and can achieve a speed range of 0~0.21 m/s by adjusting the output voltage duty cycle through a knob. Kearns CL-E100 incremental encoder is installed on the conveyor belt to measure the linear speed of the track. RealSense D415 camera captures real-time video stream images of robots and targets, SiamMask node segments and tracks targets in real-time, while Yolo_ EPnP nodes estimate the target pose in real-time. The pose of the target in the camera coordinate system is converted to the robot base coordinate system through hand eye calibration, and the PBVS control node solves the pose error between the robot end effector and the target.

Based on the position visual servo control equation, the motion speed of the end effector is solved, and the motion angles of each joint of the robot are solved through inverse kinematics. The Cartesian path planning algorithm is used to find a safe and stable motion path, and the optimal solution for the motion angle is determined. Solve the motion angles of each joint and transmit them to the robot system through the industrial Ethernet communication port to drive the robot's motion. Obtain the motion angle data of each joint of the robot teaching device, draw the joint angle curve, and obtain the error image of the two as shown in Fig. 5.

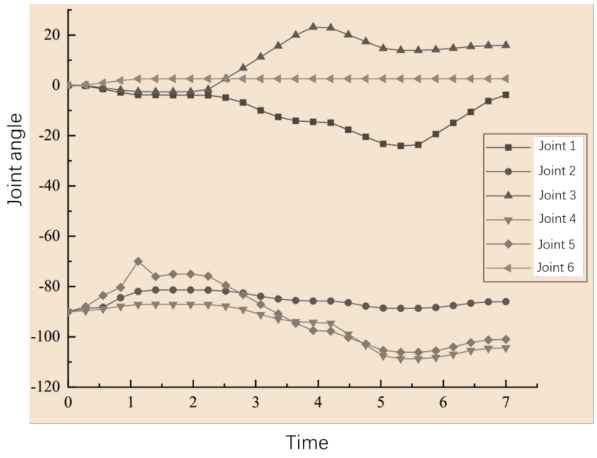

Fig. 5. Error image

This article takes the Y-axis of the robot as an example to analyze the Y-axis error curve during the grasping process, as shown in Fig. 6.

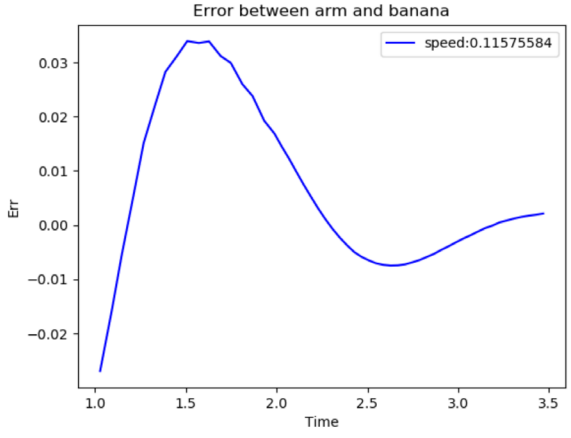

Fig. 6. Y-axis error curve

The relationship between the Y-axis position error and time from the beginning of tracking to the end of grasping can be seen that as the target object's motion speed increases, the overshoot amplitude of the error curve increases, and the robot's tracking time also increases. When the speed is 0.06m/s, the robot reaches the grasping threshold in the Cartesian tracking control algorithm, and only needs to wait for the gripper to close in the subsequent tracking process, so it can have a shorter tracking time.

6 Conclusion

This article has achieved preliminary functional design through algorithm improvement and simulation. At the same time, the following problems were found during the work process, which are also future research directions.

1) The extraction of target objects by binocular vision is not precise enough, and the accuracy of processing noise and interference needs to be further improved.

2) The tracking algorithm needs to be further optimized with a single tracking path, as this article is a single linear production line. In the future, more complex moving trajectories will be tracked and captured for research.

7 Acknowledgement

Robot Target Detection Based on Visual Servoing Control Feedback (Z23006).

References

[1] P. Wang, X.-G. Cao, H.-W. Ma, X.-D. Wu, J. Xia, Dynamic target steady and accurate grasping algorithm of gangue sorting robot based on cosine theorem-PID, Journal of China Coal Society 45(12)(2020) 4240-4247.

[2] M. Wang, J.-W. Jin, Y.-T. Cao, Dynamic target grasping control method of food sorting robot based on improved particle swarm optimization, Food & Machinery 38(3)(2022) 86-91.

[3] H.-W. Ma, N.-X. Sun, Y. Zhang, P. Wang, X.-G. Cao, J. Xia, Track planning of coal gangue sorting robot for dynamic target stable grasping, Journal of Mine Automation 48(4)(2022) 20-30.

[4] Z.-Q. Wang, H.-B. Wu, J.-H. Ye, J.-S. Xu, Dynamic catch for mobile manipulator based on Kalman filter, Journal of Mechanical & Electrical Engineering 36(8)(2019) 851-856.

[5] Z.-Y. Ge, M.-Y. Yang, Research of Visual Servo Based on ReLU Network, Computer Measurement & Control 26(8)(2018) 78-82.

[6] Q. Wan, Z. Li, Y.-K. Li, Z. Ge, Y.-N. Wang, D. Wu, Target Following Method of Mobile Robot Based on Improved YOLOX, Acta Automatica Sinica 49(7)(2023) 1558-1572.

[7] B. Yang, Z.-R. Pan, Foreground Detection Method Based on Three-frame Difference Method and Improved Vibe Algorithm, Computer & Digital Engineering 49(11)(2021) 2242-2247.

[8] W. Chen, Y. Liu, H.-T. Li, J. Sun, N. Yan, Improved ViBe algorithm based on adaptive threshold and dynamic update factor, Journal of Applied Optics 43(3)(2022) 444-452.

[9] K. Zhou, H.-B. Zhang, D.-M. Fu, Z.-Y. Zhao, H. Zeng, Design and implementation of multi-feature fusion moving target detection algorithms in a complex environment based on SiamMask, Chinese Journal of Engineering 42(3)(2020) 381-389.

[10] M. Zhu, Z. Meng, H. Zhang, Y.-Z. Sun, Trajectory Planning of 6-DOF Manipulator Based on ROS, Modular Machine Tool & Automatic Manufacturing Technique (4)(2019) 1-3.