Deep Learning-based Mobile Robot Target Object Localization and Pose Estimation Research

Caixia He[1*], Laiyun He[2]

Department of Mechanical and Electrical Engineering, Anhui Automobile Vocational and Technical College, Hefei, 230601, China[1]

Procurement Centre, Anhui Jianghuai Automobile Group Co, Ltd. Hefei, 230601, China[2]

***Abstract*—Two key technologies in robotic object grasping are target object localization and pose estimation (PE), respectively, and the addition of a robotic vision system can dramatically enhance the flexibility and accuracy of robotic object grasping. The study optimizes the classical convolutional structure in the target detection network considering the limited computing power and memory resources of the embedded platform, and replaces the original anchor frame mechanism using an adaptive anchor frame mechanism in combination with the fused depth map. For evaluating the target's pose, the smooth plane of its surface is identified using the semantic segmentation network, and the target's pose information is obtained by solving the normal vector of the plane, so that the robotic arm can absorb the object surface along the direction of the plane normal vector to achieve the target's grasping. The adaptive anchor frame can maintain an average accuracy of 85.75% even when the number of anchor frames is increased, which proves its anti-interference ability to the over fitting problem. The detection accuracy of the target localization algorithm is 98.8%; the accuracy of the PE algorithm is 74.32%; the operation speed could be 25 frames/s. It could satisfy the requirements of real-time physical grasping. In view of the vision algorithm in the study, physical grasping experiments were carried on. Then the success rate of object grasping in the experiments was above 75%, which effectively verified the practicability.**

***Keywords*—Mobile robot; target object localization; pose estimation; YOLOv2 network; FCN semantic segmentation network**

I. INTRODUCTION

There are many high-intensity and dangerous delicate operations in the actual industrial production process, and with the significant increase of labor costs in recent years, the industrial production environment requires a lot of human capital to perform these operations. For enhancing the industrial productivity and control labor costs, a lot of industrial robots are introduced in industrial environments to perform daily industrial operations [1]. The ability of robots to perform a range of complex tasks in industrial production quickly and efficiently, and with lower input costs compared to manual labor, has made them the primary choice for real-world industrial operations [2]. However, mobile robots are still very difficult to fully automate in a real-world industrial production environment, and workers are often needed to assist in the process, resulting in limited efficiency gains for the entire industrial process [3-4]. To achieve fully automated robotic operations, vision systems need to be introduced on mobile robots equipped with robotic arms [5]. The introduction of vision systems in robotics can on the one hand increase the reliability of robotic arms working in real complex industrial environments and on the other hand reduce the need for manual assistance in industrial operations [6-7]. Although a large number of mobile robots have been introduced into actual industrial production environments, they cannot fully automate actual industrial operations. Therefore, a mobile robot equipped with a robotic arm with visual feedback is needed to carry out transportation, sorting and other work in the industrial environment. To achieve this process, mobile robots first need to detect the target, locate the target position, estimate the object's posture, and determine the grasping point. The research mainly focuses on the vision algorithm of the sucking robot arm when grasping objects. The problems to be solved are target location and pose estimation. Research on combining depth information and image color information for pose estimation, and propose an adaptive anchor frame mechanism based on the characteristics of depth images. Then, the semantic segmentation network and principal component analysis are used to determine the surface normal vector of the object, in order to estimate the target pose. The purpose of the research is to make the Robotic arm adjust the pose direction of the robot arm and grasp the object more efficiently and accurately by determining the spatial position and pose of the target object.

II. RELATED WORK

Target detection is the key and prerequisite for automated object grasping by robotic arms in industrial production environments, and is therefore a research focus in machine vision. Dai Y et al. present a discriminative network for infrared small target detection to address the problem of few features inherent in purely data-driven methods, which fully utilizes labeled data and domain knowledge, and validates its performance on the open SIRST dataset, verifying that the network has some enhancement performance [8]. Scholars Szemenyei M and Estivill-Castro V present two new neural network results for the target detection problem of rescue robots in soccer tournaments, both structures use environmental attributes for enhancing the semantic segmentation and target detection, and use synthetic transfer learning to complete the learning in a small number of manually labeled images, and finally validate the models in experiments low cost and advanced [9]. Three aspects of vision-based robot grasping were investigated by Du et al. A review of traditional methods based on RGB-D image input and new methods of deep learning (DL) was mainly conducted to provide theoretical help for the challenges and solutions of robot grasping [10]. Ravindran et al. addressed the multi-target detection and multi-target tracking in vehicle driving and proposed the solution of combining sensing modalities with Deep Neural Network (DNN), which includes three sensors and fusion of sensor data with DNN, was proposed for multi-target detection and multi-target tracking problem in vehicle driving [11]. Afif et al. proposed a detection framework for specific indoor category, which is based on "RetinaNet" built and evaluated using ResNet, DenseNet and VGGNet, achieving up to 84.61% detection accuracy in the experiment [12].

After obtaining the target's position in the camera, in order to use the robotic arm to grasp the object, it is also necessary to obtain the object's pose information. Vision-based PE can be divided into two categories: learning-based PE and model-based PE. Wu et al. used linear complementary filters to deal with and depersonalize the multi-sensor PE problem in a device, specifically by obtaining a quadratic observation model through a gradient descent algorithm, and then building an additive measurement model based on the derived results, achieving a reduction in space without loss of estimation accuracy consumption and computational burden without loss of estimation accuracy [13]. Scholars Al-Sharman et al. train DNNs based on DL techniques for identifying related measurement models and filter them out, and use loss techniques to reduce computational sophistication [14]. Scholars Billings G and Johnson-Roberson M proposed SilhoNet, a new way for predicting 6D object pose in monocular camera data, which is to predict the intermediate contours of the objects with associated occlusion masks and 3D translation vectors, and then regress 3D orientation from the contours, obtaining better experimental performance than two networks Estimation performance [15]. Wang et al. presented a DL-based grasping pose estimation method for a SCARA loading and unloading robot, which fuses point clouds with category numbers into a point category vector and uses multi-point mesh networks for evaluating the robot's grasping pose, getting success rates of 98.89%, 98.89%, and 94.44% on three homemade sub-datasets [16]. Liu et al. proposed a grasping posture determination method related to shape analysis for target object shape analysis in robotic grasping, which reduces complicated objects to basic shapes and then simplifies the grasping of objects based on force closure [17].

Comprehensive domestic and international research on mobile robot target detection and PE reveals that most of the detection algorithms are related to DL, which is computationally intensive, while the learning-based PE methods also rely heavily on the diversity of training data sets, which requires high data collection and calibration. Therefore, the study reduces the computational effort of target detection in the embedded platform by optimizing the original convolution process, and then performs PE by the Fully Convolution Network (FCN) semantic segmentation and (Principal Component Analysis (PCA) algorithm, aiming to provide a more concise and practical mobile robot vision algorithm.

III. Target Object Localization Algorithm and Pose Estimation Algorithm for Mobile Robot

A. *Target Localization Algorithm and Optimization Based on YOLOv2 Network*

Based on the progress of computer technology and artificial intelligence technology, the robotics industry has also developed rapidly, and robots have been applied to more fields, especially in tasks with harsh working conditions and strong repeatability. Using robots to perform these tasks can liberate workers from harsh working environments and also improve work efficiency. In many robot work scenarios, the most common action performed by robots is grasping. Robots perceive the surrounding environment through sensors and then perform grasping operations. When a mobile robot performs grasping of a target object, it must obtain the correct object position and pose to ensure that the robot arm accurately grasps the target from a suitable position and with the correct grasping pose. That is, there are two important problems to be solved in the whole grasping process: localization of the target object and estimation of the spatial pose of the target. The study uses computer vision algorithms to solve the problems faced by mobile robots performing industrial production operations, and the specific process is shown in Fig. 1.

Neural networks have powerful feature extraction capabilities, and with a sufficient number of training datasets with labels, the gradient back-propagation algorithm can be used to renew the weights of the neural network to achieve the coordinate position detection of different target objects. The YOLOv2 network is in view of the Darknet network. It has a powerful feature extraction capability and uses the anchor frame mechanism instead of the direct regression of the target frame coordinates in YOLOv1. However, although the YOLOv2 network has a relatively small model structure and fast localization detection speed, it is still difficult to be arranged in related platforms with very limited resources, so the network structure needs to be further optimized for decreasing the model's size. Fig. 2 indicates the structure of the convolutional layer.

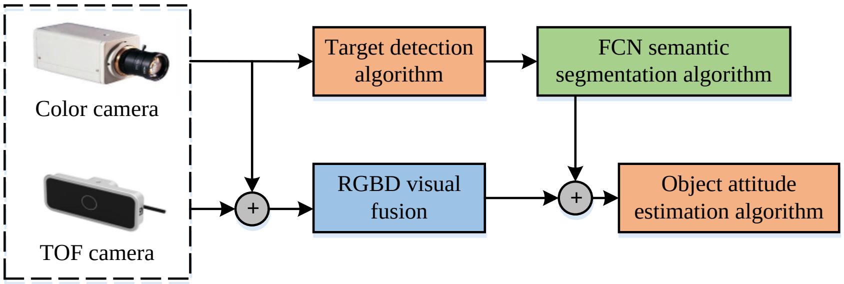
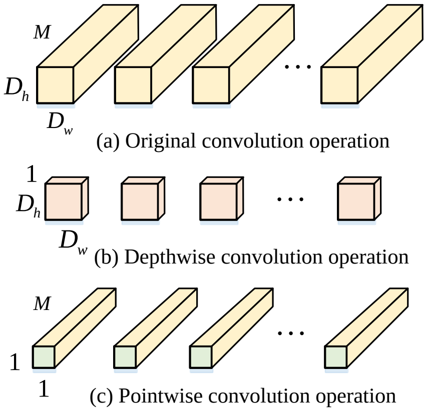

Color camera
TOF camera
Target detection algorithm
FCN semantic segmentation algorithm
RGBD visual fusion
Object attitude estimation algorithm
+
+

Fig. 1. Computer vision algorithm flowchart.



(a) Original convolution operation

(b) Depthwise convolution operation

(c) Pointwise convolution operation

Fig. 2. Structure diagram of convolution layer.

It supposes that a feature map (FM) of $D_W \times D_H \times M$ is input and a feature map of $D_W \times D_H \times N$ is output in the standard convolution operation (CO) as in Fig. 2(a), where D_H and D_W are the length and width of the FM, respectively, and M serves as the channels' quantity of the input FM and N serves as the channels' quantity of the output FM. Assuming that the convolution kernel's (CK) size is D_k and the step size is 1, the parameters quantity of the CK is $D_k \times D_k \times M \times N$, then the amount of computation generated by one CO is shown in Equation (1).

$$D_k \times D_k \times M \times N \times D_W \times D_H \qquad (1)$$

The standard CO is divided into two processes: filtering and combining. For decreasing the size of the network, the standard CO is split into depthwise convolution, which is only responsible for filtering, and pointwise convolution, which is only responsible for combining. The depth wise convolution in Fig. 2(b) uses a single-channel CK on each channel of the input FM to generate corresponding feature values at each position on each channel of the input FM. Then the point wise CO is used, i.e., a 1 × 1 CK is used to combine the feature values on different channels at the same position to produce the corresponding feature vectors. Compared with the standard CO, the parameters quantity for depth wise convolution is $D_W \times D_H \times M$ and the parameters quantity for point wise convolution is $1 \times 1 \times M \times N$. Equation (2) demonstrates the parameters quantity for the two-step CO.

$$D_k \times D_k \times M + M \times N \qquad (2)$$

And the two-step CO produces the computation as shown in Equation (3).

$$D_k \times D_k \times M \times D_W \times D_H + M \times N \times D_W \times D_H \qquad (3)$$

Compare the transport arithmetic before and after splitting the standard CO into two parts, depth wise CK and point wise convolution, as shown in Equation (4).

$$\frac{D_k \times D_k \times M \times D_W \times D_H + M \times N \times D_W \times D_H}{D_k \times D_k \times M \times N \times D_W \times D_H} = \frac{1}{N} + \frac{1}{D_k^{\ 2}} \qquad (4)$$

The size of CK is usually assumed to be 3, so the former term in Equation (4) can be neglected, i.e., by splitting the standard CO, the number of CK parameters can be reduced while the computation is reduced to one-ninth of the standard CO. In order to facilitate more accurate target detection and localization by the machine, an adaptive anchor frame mechanism is presented to obtain the 3D position of the object by using additional depth pictures to complement the information of the color pictures. The adaptive anchor frame mechanism only requires pre-setting n anchor frames with different aspect ratios of 1. The width and height of the anchor frames are multiplied by the scale factor calculated from the depth image to obtain the effect of the original anchor frame. The YOLOv2 network framework after adding the adaptive anchor frames is shown in Fig. 3.

Fig. 3 illustrates that the input image is subjected to the YOLOv2 network to generate the prediction parameter, which is used to improve the shape of the anchor frame and produce the normalized prediction frame. The depth image is processed to obtain the scale factor map, and the final detection result is obtained by multiplying the scale factor corresponding to each pixel to the prediction frame. When the camera captures an object, the same object has different distances from the camera and has different sizes in the computer's field of view, thus the object size can be obtained by combining the depth fusion map. The correspondence between depth distance and object size is shown in Fig. 4.

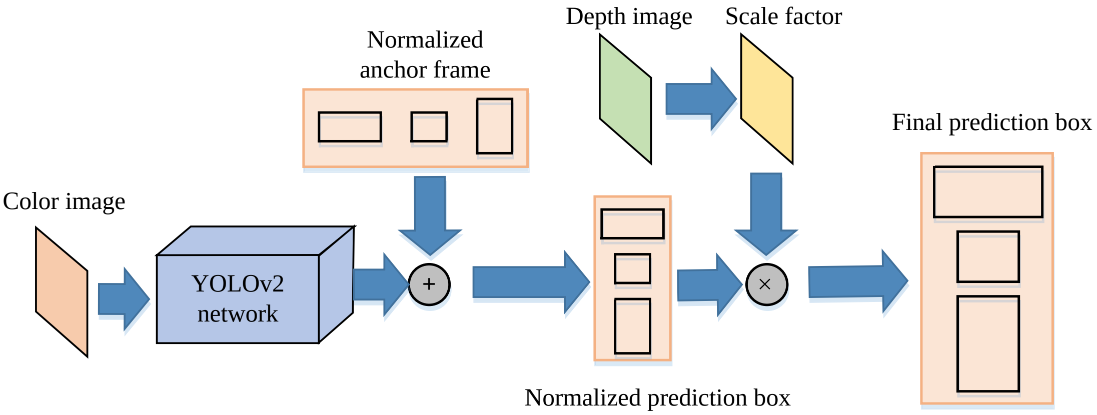


Fig. 3. YOLOv2 network combined with adaptive anchor frame mechanism network block diagram.

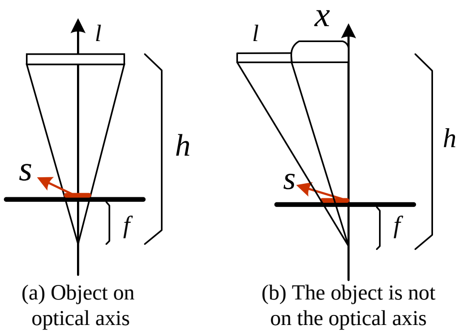

l
h
s
f
x
l
h
s
f
(a) Object on optical axis
(b) The object is not on the optical axis

Fig. 4. Correspondence between depth distance and object size.

Fig. 4(a) indicates that the size of the object imaged on the imaging plane of the camera is s can be calculated based on the principle of similar triangles, as shown in Equation (5).

$$s = \frac{f \times l}{h} \tag{5}$$

In Equation (5), l is the actual length of the object, h serves as the distance between the object and the camera, and f serves as the focal length of the camera. In Fig. 4(b), the image size can also be calculated as shown in Equation (6).

$$s = \frac{f \times (l + x)}{h} - \frac{f \times x}{h} = \frac{f \times l}{h} \tag{6}$$

The size factor can be approximated based on this model of the relationship between the object-to-camera distance and the imaging size on the imaging plane, and multiplied by the normalized anchor frame to achieve the original anchor frame effect and get a relatively accurate prediction frame. For fully utilizing the information of all the prediction frames generated on the same object, a soft NMS is used in the study specifically by doing a weighted average of the coordinate information of all the frames to get the final prediction frame Box_i , as shown in Equation (7).

$$Box_i = \frac{\sum_j conf_{ij} \times box_{ij}}{\sum_i conf_{ij}} \tag{7}$$

In Equation (7), box_{ij} is the j th predictor box output on the i th object, and $conf_{ij}$ is the confidence score of the predictor box. Finally, the overfitting phenomenon caused by limited training samples is solved by training the anchor frame parameters in steps. When training the anchor frames individually, the training data assigned to each anchor frame almost doubles, thus overcoming the over fitting phenomenon of a single anchor frame due to insufficient training samples.

B. *Target Object Pose Estimation Algorithm*

In actual industry and life, mobile robots often need to grasp objects with various shapes, uncertain postures, and possible occlusion between objects. Therefore, it is necessary to obtain the position and attitude information of the target object through appropriate methods, and then use a robotic arm to grasp the target object. After the position of the object in the camera is determined by the target detection and localization algorithm, the spatial coordinate values of the object can be obtained by using the camera's internal reference and related fused depth maps. However, the robotic arm also needs to know whether there is a plane on the object surface that can be absorbed when it grasps the object. This process can be done by semantic segmentation network to do pixel-level classification of the pixel points on the object surface and extract the points on the object surface that can be grasped, and the maximum connected domain (CD) consisting of these points is the absorbable plane (AP). The plane normal vector (PNV) of the plane equation in the 3D space established by these points is the pose direction of the target object, while the center on the CD is chosen as the target's 3D spatial location. The flow of the PE algorithm in the study is shown in Fig. 5.

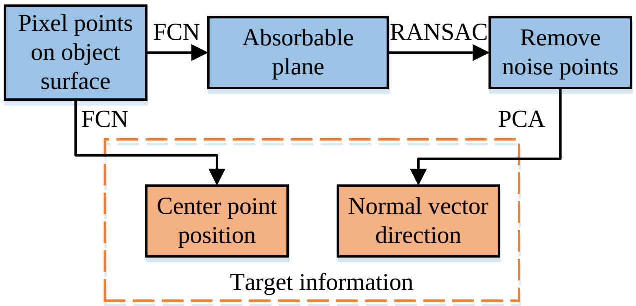


Fig. 5. Flow chart of attitude estimation algorithm.

In Fig. 5, the PE first extracts its AP using the FCN neural network algorithm, then removes the noisy points of the plane using the Random Sampling Consensus (RANdom SAmple Consensus, RANSAC), and finally solves the parameters of the object surface model using PCA for getting the target's pose message. Since there is noise in the depth map by the camera, after using FCN to determine the joint area on the AP of the target, RANSAC is used for removing the noise with large errors before getting a more accurate plane model. The processing flow is shown in Fig. 6.

The planar model used in the study has four parameters. Therefore, four data points (DP) are required for addressing the model. In Fig. 6, the RANSAC algorithm randomly selects four DP in the data set generated from the FCN results for solving the model parameters. All DP are included in the solved model, and the statistical error is less than the internal points' quantity. The model is considered accurate only when the internal points' quantity exceeds the set threshold, and then the PCA algorithm is used for addressing the related model parameters, and if the error of the current optimal model is greater than that of the obtained model, the optimal model is renewed.

The PCA algorithm is used to compress the data in the original feature space into a lower dimensional space. A schematic diagram of the PCA algorithm and its solution to the PNV is shown in Fig. 7.

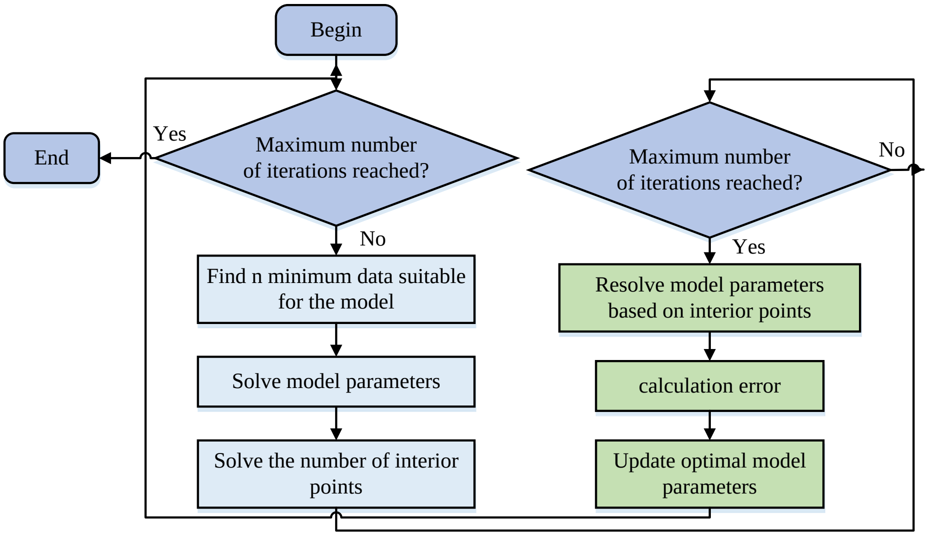
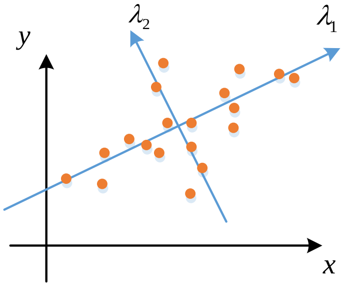
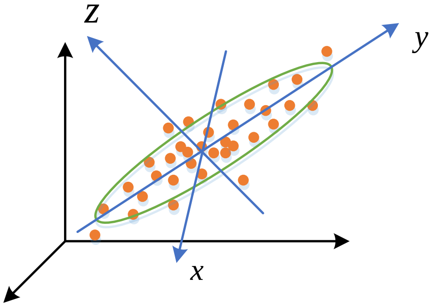

Begin
Maximum number of iterations reached?
Yes
End
No
Find n minimum data suitable for the model
Solve model parameters
Solve the number of interior points
Maximum number of iterations reached?
No
Yes
Resolve model parameters based on interior points
calculation error
Update optimal model parameters

Fig. 6. Flow chart of RANSAC algorithm.



(a) Schematic diagram of PCA algorithm principle



(b) Schematic Diagram of PCA Algorithm for Solving Plane Normal Vector

Fig. 7. PCA algorithm and its schematic diagram for solving plane normal vector.

In Fig. 7(a), PCA transforms the data in the original feature space into the new orthogonal feature space by linear transformation, and then removes the dimensional information that is less informative, leaving a number of dimensions with more informative information to express the original data. The amount of information in a dimension can be expressed by the variance of the data in that dimension; the larger the variance, the greater the amount of information. The projection transformation is shown in Equation (8).

$$\begin{cases}\delta = A^T x \\ A = \left[\alpha_1, \alpha_2, \cdots, \alpha_n\right]\end{cases} \tag{8}$$

In Equation (8), x is the data in the dataset, A is the change transformation matrix, the projected data is δ , and the feature space formed by α as the feature vector is the projected space where the vectors have the relationship shown in Equation (9).

$$\begin{cases}\alpha_1 \alpha_i = 1 \\ \alpha_i \alpha_j = 0, i \neq j\end{cases} \tag{9}$$

The data in the original feature space is projected into the feature space consisting of α as the feature vector by the projection transformation, and the projection value of the vector x in α_i δ_i is shown in Equation (10).

$$\delta_i = a_i^T x \tag{10}$$

The principle of PCA is to let the information in the dataset fall into the feature space as much as possible, so the variance of the projection on the new feature vector should also be as large as possible. The data variance of the projected data in one dimension is shown in Equation (11).

$$\begin{aligned} D\left(\delta_i^2\right) &= E\left(\delta_i^2\right) - E^2\left(\delta_i\right) \\ &= E\left(\alpha_i^T x x^T \alpha_i\right) - E\left(\alpha_i^T x\right) E\left(x^T \alpha_i\right) \\ &= \alpha_i^T \sum \alpha_i \end{aligned} \tag{11}$$

In Equation (11), $D\left(\delta_i^2\right)$ is the variance after projection. To maximize it and to satisfy the relation, it is solved using the Lagrange multiplier method as shown in Equation (12).

$$f\left(x\right) = \alpha_i^T \sum \alpha_i - \lambda\left(\alpha_i^T \alpha_i - 1\right) \tag{12}$$

In Equation (12), λ serves as the eigenvalue of the matrix Σ and α_i serves as the corresponding eigenvector. The derivative of α_i is obtained when the derivative is 0. The maximum value of $D\left(\delta_i^2\right)$ is obtained when the derivative is 0, as shown in Equation (13).

$$\begin{cases} \dfrac{\partial f\left(x\right)}{\partial a_i} = \Sigma\alpha_i - \lambda\alpha_i \\ \Sigma a_i = \lambda\alpha_i \end{cases} \tag{13}$$

Since the points on the AP of the target object are distributed in the whole plane space, the two eigenvectors along the plane direction have the largest variance, and the eigenvector normal to the plane direction corresponds to the smallest eigenvalue, so the eigenvector corresponding to the smallest eigenvalue found by PCA is the normal vector of the AP. With the obtained PNV as the target's pose direction, the grasping of the robot arm for the target can be realized. As shown in Fig. 6(b), PCA first determines the two feature vectors with the largest variance x and y , and determines the z axis direction in view of certain premises. Due to the small impact of sensor noise on the z axis direction, a portion of the sensor noise is successfully filtered out using the PCA method. The evaluation metric for the target PE is the 2D projection metric, and the PE is considered accurate if the average distance between the projection of the predicted corner point and the real labeled corner point e_{REF} is less than 5 pixels. 2D reprojection metric is defined as shown in Equation (14).

$$e_{REF} = \left\| P_i - TM\mu \right\|_2 \tag{14}$$

In Equation (14), M_C is the camera matrix, G is the target pose to be estimated, P_i is the position of the i th pixel, and μ is the average of the pixel distribution with the maximum blending weight.

IV. Analysis of the Effect of Target Object Localization and Pose Estimation for Mobile Robots

A. *Performance of Target Object Localization Algorithm for Mobile Robots*

The study is based on a mobile robot platform to test the visual perception algorithm, including the performance analysis of target localization algorithm, PE algorithm and the effect analysis of the robot arm's grasping for the target. The mobile robot platform is equipped with a robot arm system, a color depth binocular vision system, a TX2 DL IPC and an image acquisition IPC for completing the grasping process of the target. The initial Learning rate of network training is 0.001, and the Learning rate of every 100 epochs is divided by 10. The configuration parameters of the experimental hardware are shown in Table I.

The public dataset used in the object detection and positioning experiment is from LineMod, which is a standard dataset for attitude estimation. There are 1200 instances of 13 objects, and the data includes color maps, depth maps, and corresponding camera coordinate information from different perspectives. In order to improve operational efficiency, the study selected 200 images of each of the four types of objects for comparative analysis of different anchor box mechanisms and to compare the performance of the algorithms before and after the improvement. The mean Average Precision (mAP) results of the original anchor frame mechanism and the adaptive anchor frame mechanism in YOLOv2 are shown in Fig. 8.

In Fig. 8, the accuracy of the adaptive anchor frame mechanism improves by 1.55% when there are only 1 or 2 anchor frames, and the improvement is more obvious. When the number of anchor frames is three or more, the original anchor frame mechanism shows a serious over fitting phenomenon, and the detection accuracy decreases by 3.65%~3.77%. The adaptive anchor frame mechanism can still maintain a high detection accuracy when the number of anchor frames is three and four, which indicates that it has some improvement effect on the over fitting problem. The detection accuracies of different target detection and localization algorithms are shown in Fig. 9.

TABLE I. Configuration of Experimental Hardware Parameters

Configuration	TX2 Deep Learning Industrial Control Board	Image acquisition industrial control board
CPU	ARM Contex-A57	Intel Bay Trail J1900
Memory	8GB LPDDR4	8G DDR3L 1333MHz
Hard disk	32GB eMMC5.1	64GB Solid-state drive
interface	Wireless, Bluetooth, Ethernet	Network interface, serial port, USB

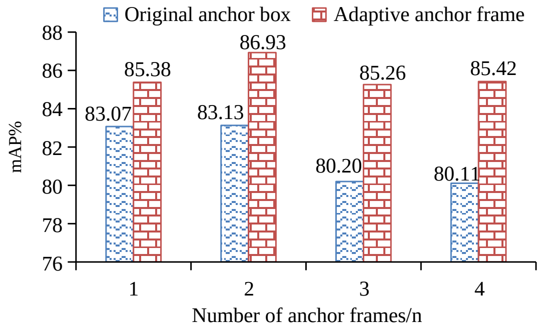


Fig. 8. Comparison of mAP results between the original anchor frame mechanism and the adaptive anchor frame mechanism under different number of anchor frames frames.

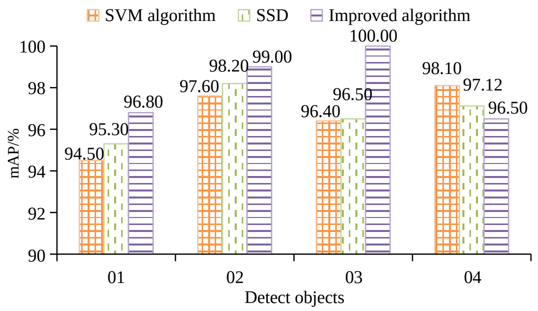


Fig. 9. Detection accuracy of target detection and location algorithm before and after improvement.

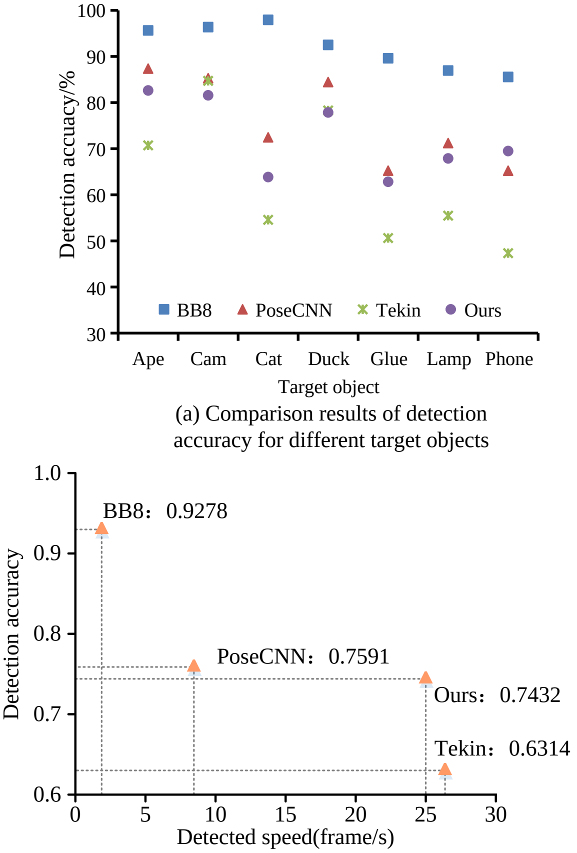
In Fig. 9, the improved algorithm detects objects 01, 02, and 03 significantly better than the Support Vector Machine (SVM) algorithm and the Single Shot multiBox Detector (SSD) algorithm, and the detection accuracy for object 04 is lower than the other two algorithms, but still above 95%. The SSD algorithm has good detection speed and accuracy compared to the SVM algorithm, but it is still weak in detecting small object 02. The average detection accuracies of the improved algorithm, SVM algorithm and SSD algorithm for objects are 98.8%, 96.65% and 96.78%, respectively. Taken together, the YOLOv2 network used in the study has high detection accuracy, good small object detection ability, and the optimized model size can be applied to embedded platforms, which is the optimal choice.

B. Performance of Target Object Pose Estimation Algorithm for Mobile Robots

The training samples for the pose estimation network model for mobile robots are taken from the LineMod dataset. In order to better simulate the real work environment and verify the stability of the algorithm, the dataset used during the testing was the Occlusion LineMod dataset, which was reannotated and generated from the LineMod dataset, was used during testing. This dataset contains 1435 images of eight objects with complex backgrounds and occlusions. For testing the PE algorithm in the study, it is compared with several commonly used PE algorithms for experiments. The PE results of different algorithms for seven target objects are shown in Fig. 10.

In Fig. 10(a), the BB8 algorithm has the highest PE accuracy for seven types of targets, followed by the PoseCNN algorithm, and the detection accuracy of the proposed PE algorithm is close to that of PoseCNN. However, according to the average detection accuracy and detection speed in Fig. 10(b), although the detection accuracy of BB8 is 92.78%, its detection speed is only 2 frames/s.

The detection accuracy of PoseCNN is 75.91% and the detection speed is 7 frames/s, which is slightly higher than that of BB8. The detection accuracy of PoseCNN is 75.91% and the detection speed is 7 frames/s, which is slightly higher than that of BB8. The Tekin algorithm runs the fastest at 26 frames/s, but its estimation accuracy is only 63.14%. The accuracy of the PE algorithm is 74.32%, which is a big improvement over Tekin's algorithm, and it runs at 25 fps, which seems to satisfy the needs of real-time operation. Table II depicts the experimental results of the visual perception algorithm for different objects in the real object grasping experiments.



(a) Comparison results of detection accuracy for different target objects

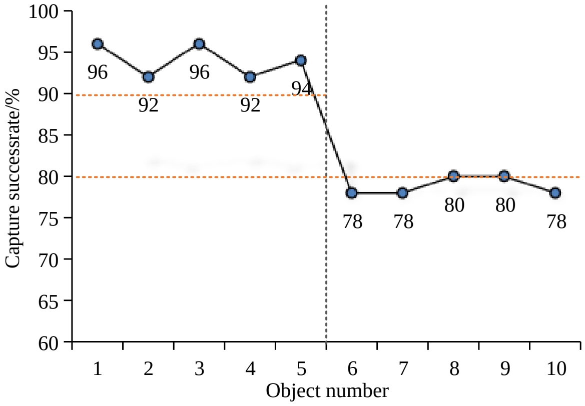
(b) Comparison results of detection accuracy and speed of different algorithms

Fig. 10. Comparison of results of common object attitude estimation algorithms.

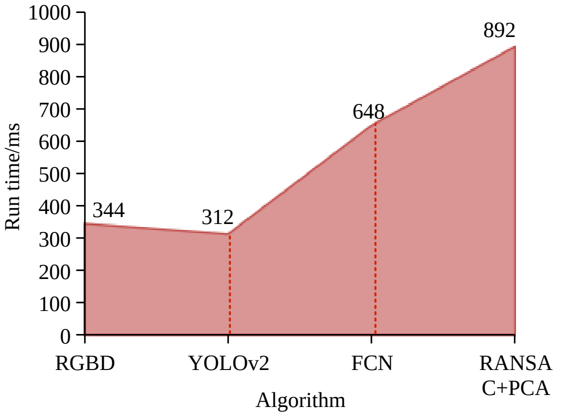
TABLE II. RESULTS OF VISUAL PERCEPTION ALGORITHM ON DIFFERENT OBJECTS

Object number	Surface center coordinate point	Coordinate point directly above	Surface normal vector	Center pixel coordinates	Pixel coordinates directly above
1	(0.09,0.24,1.84)	(0.12,0.24,1.64)	(-0.17,0.00,0.99)	(710,665)	(751,688)
2	(-0.06,0.11,1.36)	(-0.07,0.15,1.39)	(0.03,0.87,0.25)	(621,537)	(614,385)
3	(-0.10,0.07,1.54)	(-0.11,0.12,1.51)	(0.04,1.00,0.14)	(556,548)	(543,367)
4	(0.00,0.14, 1.50)	(0.06,-0.05,1.45)	(-0.28,0.95,0.17)	(646,608)	(708,434)
5	(0.05,0.21,1.18)	(0.05,0.31,1.01)	(-0.04,0.50,0.87)	(699,727)	(722,906)
6	(0.07,0.06,1.27)	(0.08,-0.12,1.19)	(-0.03,0.91,0.42)	(721,546)	(734,337)
7	(0.08,0.05,1.37)	(0.09,-0.13,1.28)	(-0.03,0.89,0.45)	(728,532)	(740,340)
8	(0.06,0.16,1.35)	(0.07,0.27,1.18)	(-0.04,-0.52,0.86)	(710, 649)	(728,796)
9	(0.07,0.04,1.28)	(0.08,-0.11,1.37)	(-0.03,0.82,0.49)	(725,518)	(746,322)
10	(-0.08,0.06,1.14)	(-0.09,0.08,1.46)	(0.05,1.01,0.15)	(558,543)	(547,361)

In the physical object grasping experiments, a total of 10 unknown objects were grasped, with object numbers 1 to 5 for normal-sized objects and 6 to 10 for smaller-sized objects. In Table I, the surface center coordinate point indicates the three-dimensional spatial point of the absorbable point on the object's surface in the corresponding coordinate system, and the coordinate point directly above indicates the three-dimensional spatial point at 20 cm directly above the center coordinate point. The robot arm system controls the end of the robot arm to move to the upper coordinate point, and adjusts the direction of the end nozzle to be consistent with the PNV, and then makes it move to the surface center coordinate point along the normal direction for completing the grasping of the target. The pixel coordinates projected to the color camera coordinate system are the center pixel point and the upper pixel point. The results of the grasping success rate and the each algorithm's time are shown in Fig. 11 for 50 grasps of each object.



(a) Success rate result of grasping target object



(b) Running time of each algorithm in the process

Fig. 11. The success rate of grasping objects and the running time of each algorithm in the process.

In Fig. 11(a), the success rates of physical grasping for objects 1~5 of normal size are all over 90%, while the success rates of physical grasping for objects 6~10 of smaller size are reduced but still maintain around 80%. The average success rate of physical object grasping reaches 86.4%, which tests the practicality of the physical object grasping algorithm proposed in the study. The study also tested the running time of the vision algorithm for each stage. In Fig. 11(b), although the neural network algorithm is computationally intensive, it does not account for a large percentage of the total algorithm running time because the target detection algorithm runs on the GPU and the optimization of the DL framework substantially increases the neural network's speed. The RANSAC algorithm takes up the largest percentage of the time because it requires multiple iterations and the iterative process also uses PCA to calculate the interior point error.

V. Conclusion

As robots are used to replace tedious manual labor in more and more industries, the use of mobile robots to complete the handling of goods in the logistics industry has gradually become a hot research topic nowadays. The study designs a set of vision algorithms for a mobile robot platform for the vision system of fully automated handling, mainly including a target object detection and localization algorithm based on the embedded platform with improved convolutional structure and an object PE algorithm based on FCN semantic segmentation network. While the detection accuracy of the original anchor frame mechanism decreases by 3.65%~3.77% due to the overfitting phenomenon, the proposed adaptive anchor frame mechanism can still maintain a high detection accuracy with good resistance to overfitting when the number of anchor frames is 3 and 4. In the experiments of detection and localization of different objects, the target localization algorithm proposed in the study improves the detection accuracy by 2.22% and 2.09% compared with the SVM algorithm and the SSD algorithm, respectively, with better localization results. The average success rate of grasping physical objects also reaches 86.4%, which effectively tests the algorithm's practicality proposed in the study for physical object grasping. However, although the study has optimized the convolutional structure and reduced the network's model parameters, the computational burden is still too large for the embedded platform, and the base convolutional layers can be considered to be combined together in subsequent studies to further reduce the model size of the network.

VI. Discussion and Prospects

The study used object detection networks to determine the three-dimensional position information of objects and semantic segmentation networks to assist in estimating the pose of objects. Although the research has optimized the convolution structure of the network, reduced the model parameters of the network, and improved the operation efficiency of the feedforward network, for the embedded platform, the computational burden of using two convolutional neural networks is still too large, resulting in the overall operation efficiency of the system is not very ideal. Jiang D et al. used an improved Fast RCNN to achieve tasks such as semantic segmentation, object classification, and detection in indoor scenes, resulting in a model with good performance and high efficiency [18]. Scholar Feng T used Mask RCNN combined with a single multi box detector algorithm to achieve gesture detection and recognition in human-computer interaction, which has high detection accuracy and speed [19]. Therefore, in future research, it can be considered to draw on the solutions of these two networks and merge the basic convolutional layers together to reduce repetitive operations in the network. The results of the target detection network can also be projected onto the intermediate feature map, and the FCN head network can be run on the extracted feature image pixels to further improve the running speed of the feed forward network.

References

[1] Z. B. Li, S. Li, and X. Luo, “An overview of calibration technology of industrial robots,” IEEE/CAA J. Autom. Sinica, vol. 8, no. 1, pp. 23-36, Jan. 2021.

[2] H. Cheng, R. Jia, D. Li, and H. Li, “The rise of robots in China,” J. Econ. Perspect., vol. 33, no. 2, pp. 71-88, 2019.

[3] T. Chen, X. Liu, B. Xia, W. Wang, and Y. Lai, “Unsupervised anomaly detection of industrial robots using sliding-window convolutional variational autoencoder,” IEEE Access, vol. 8, pp. 47072-47081, Mar. 2020.

[4] T. Brito, J. Queiroz, L. Piardi, L. A. Fernandes, J. Lima, and P. Leitão, “A machine learning approach for collaborative robot smart manufacturing inspection for quality control systems,” Procedia Manuf., vol. 51, pp. 11-18, Nov. 2020.

[5] A. I. Martyshkin, “Motion planning algorithm for a mobile robot with a smart machine vision system,” Nexo, vol. 33, no. 2, pp. 651-671, 2020.

[6] R. Zeng, Y. Wen, W. Zhao, and Y. J. Liu, “View planning in robot active vision: a survey of systems, algorithms, and applications,” Comput. Vis. Media, vol. 6, pp. 225-245, Aug. 2020.

[7] A. Kazemian, X. Yuan, O. Davtalab, and B. Khnshnevis, “Computer vision for real-time extrusion quality monitoring and control in robotic construction,” Automat. Constr., vol. 101, pp. 92-98, May. 2019.

[8] Y. Dai, Y. Wu, F. Zhou, and K. Barnard, “Attentional local contrast networks for infrared small target detection,” IEEE Trans. Geosci. Remote Sen., vol. 59, no. 11, pp. 9813-9824, Nov. 2021.

[9] M. Szemenyei and V. Estivill-Castro, “Fully neural object detection solutions for robot soccer,” Neural Comput. Appl., vol. 34, no. 24, pp. 21419- 21432, Dec. 2022.

[10] G. Du, K. Wang, S. Lian, and K. Zhao, “Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: A review,” Artif. Intell. Rev, vol. 54, no. 3, pp. 1677-1734, Mar. 2021.

[11] R. Ravindran, M. J. Santora, and M. M. Jamali, “Multi-object detection and tracking, based on DNN, for autonomous vehicles: A review,” IEEE Sens. J., vol. 21, no. 5, pp. 5668-5677, Mar. 2021.

[12] M. Afif, R. Ayachi, Y. Said, E. Pissaloux, and M. Atri, “An evaluation of retinanet on indoor object detection for blind and visually impaired persons assistance navigation,” Neural Process. Lett., vol. 51, pp. 2265-2279, Jun. 2020.

[13] J. Wu, Z. Zhou, H. Fourati, R. Li, and M. Liu, “Generalized linear quaternion complementary filter for attitude estimation from multisensor observations: An optimization approach,” IEEE Trans. Autom. Sci. Eng., vol. 16, no. 3, pp. 1330-1343, Jul. 2019.

[14] M. K. Al-Sharman, Y. Zweiri, M. A. K. Jaradat, R. AI-Husari, D, D. Gan, and L. Seneviratne, “Deep-learning-based neural network training for state estimation enhancement: Application to attitude estimation,” IEEE Trans. Instrume. Meas., vol. 69, no. 1, pp. 24-34, Jan. 2020.

[15] G. Billings and M. Johnson-Roberson, “Silhonet: An RGB method for 6d object pose estimation,” IEEE Robot. Automat. Lett., vol. 4, no. 4, pp. 3727-3734, Oct. 2019.

[16] Z. Wang, Y. Xu, Q. He, Z. Fang, G. Xu, and J. Fu, “Grasping pose estimation for SCARA robot based on deep learning of point cloud,” Int. J. Adv Manuf. Technol., vol. 108, pp. 1217-1231, May. 2020.

[17] Y. Liu, D. Jiang, B. Tao, J. Qi, G. Jiang, J. Yun, L. Huang, X. Tong, B. Chen, and G. Li, “Grasping posture of humanoid manipulator based on target shape analysis and force closure,” Alexandria Eng. J., vol. 61, no. 5, pp. 3959-3969, May. 2022.

[18] Jiang D, Li G, Tan C, Hunag L, Sun Y, Kong J, “Semantic segmentation for multiscale target based on object recognition using the improved Faster-RCNN model,” Future Generation Computer Systems, vol. 123, pp. 94-104, Oct. 2021.

[19] T. Feng, “Mask RCNN-based single shot multibox detector for gesture recognition in physical education,” J. Appl. Sci. Eng., vol. 26, no. 3, pp. 377-385, Jun. 2022.

[20] X. Nie, M. Duan, H. Ding, B. Hu, and E. K. Wong, “Attention mask R-CNN for ship detection and segmentation from remote sensing images,” IEEE Access, vol. 8, pp. 9325-9334, Jan. 2020.