

# Visual recognition of robot targets in complex state based on sub-pixel Harris corner

Xinyuan Ying<sup>1</sup>, Ziyou Zhang<sup>1</sup>

<sup>1</sup>Zhejiang University of Technology, Hangzhou, Zhejiang, China

**Abstract.** In the field of robotics, it is always a big challenge for the visual recognition and operation of target objects in complex state, such as target objects in dead corner and surrounded by other targets. In this paper, V-REP and MATLAB are used for joint simulation to conduct experiments on the robot scene. For the target object in a complex state, the RGBD camera is used to record the image and determine the target range, and introduce sub-pixel Harris corner detection to establish the grasping surface and center point coordinates, to make the robot under complex condition can more accurately for corresponding operation.

## 1 Introduction

In the field of robotics, machine vision technology has very strong technical advantages in target recognition and target pose determination. It has the characteristics of large amount of information, high accuracy, non-contact measurement, and fast response [1], and has been widely used in robots Application in various fields. However, for some complex conditions, such as the target being in a closed environment or partially obscured by other objects, it often brings greater difficulty for machine vision recognition and robots to implement corresponding automatic operations. We use V-REP and MATLAB software[2] to visually recognize and process the target objects in the scene.

## 2 Experimental platform

### 2.1. Joint simulation of V-REP and MATLAB

As a three-dimensional motion simulation software, V-REP itself can perform general modeling. It also has the advantages of open source and rich API. It can be combined with MATLAB and other software for co-simulation, but MATLAB is difficult to effectively perform the robot and its working environment. Three-dimensional simulation. Co-simulation can not only use the powerful computing power of MATLAB, but also make the three-dimensional scene graph vividly displayed in V-REP [2]. We built a communication interface between V-REP and MATLAB so that the robot model in V-REP can respond to MATLAB's control commands.

### 2.2. Introduction of Youbot Robot

For the robot platform, we use Youbot robot. Youbot robot is a high-performance multi-degree-of-freedom compound robot produced by KUKA in Germany, which can realize functions such as autonomous navigation and movement and target grabbing. It consists of an omnidirectional movable chassis, a 5-degree-of-freedom manipulator, a single-degree-of-freedom opening and closing hand grip, an RGBD depth camera, and a lidar. The composition of the robot system and the configuration of the robot arm are shown in the figure below.

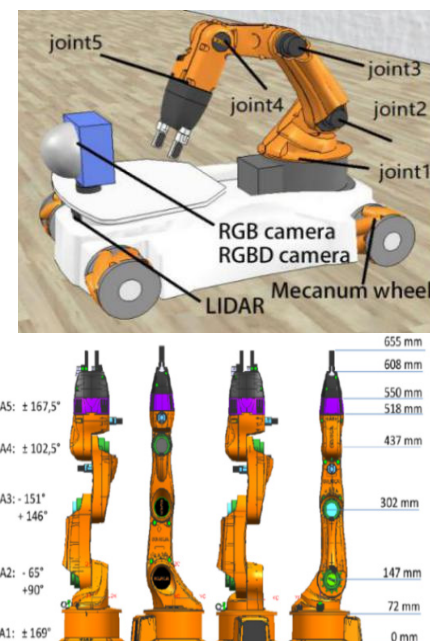


Figure 1. Youbot robot system composition and robotic arm configuration

\* Corresponding author: 201806061420@zjut.edu.cn

In order to make the youbot robot have the corresponding vision module function, we equip the mobile robot with an RGBD camera, which can obtain the target image information and the position information relative to the coordinate system at the same time, and thus control the robot's movement and perform the corresponding manipulator operation.

### 3 Image acquisition and target recognition

#### 3.1. Image acquisition platform

The RGBD camera in the robot platform consists of a color camera RGB sensor with a resolution of 512\*512 and a range-finding radar XYZsensor with a resolution of 32\*32. While the image information is obtained by the color camera, the relative distance can be obtained by the range-finding radar. Based on the world coordinate system and the position information of the object, and feedback to the platform to establish a corresponding grasping strategy. In V-REP we set up a simulation scene and set the RGBD camera at the position where the coordinates [0,0,1m] are facing downward. The simulation scenario and the results of the joint simulation with MATLAB are shown in the figure below.

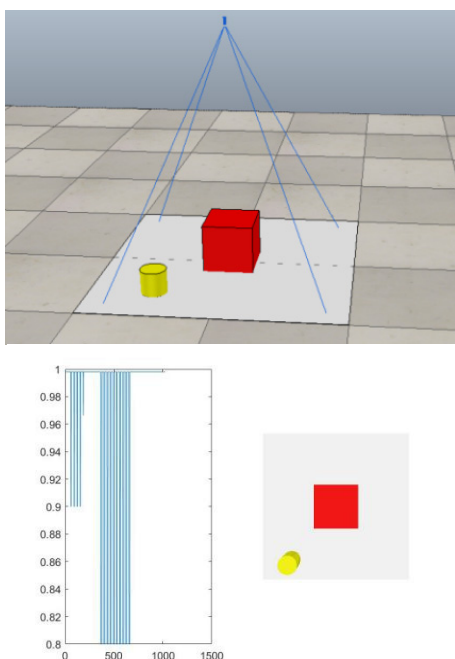


Figure 2. RGBD camera demo scene and simulation recognition results

#### 3.2. Target recognition

In MATLAB, we mainly obtain the image in V-REP by obtaining the handle of the object, and save the image in the matrix. The matrix represents the corresponding numerical output after digital image processing, and each element corresponds to a corresponding RGB value. Digital image processing is essentially an A/D processing process. Through image sampling and

quantization, when the image in vrep is transferred to MATLAB, the conversion from analog light to digital output numerical matrix can be realized.

For color images, after the digital image is transferred to MATLAB, each element in the three matrices stores the corresponding red, green, and blue RGB values, thereby recording the corresponding color and brightness information. By presetting the RGB value of each color type in the computer, we can extract the corresponding color characteristics. Similarly, we can also pre-enter the corresponding target through the camera to obtain a more accurate RGB value of the target.

In V-REP, we set up a complex situation simulation scene. The target object is a red cuboid, and it is in the scene between plants, dog models and other objects in the scene, and is located at the corner of the wall, so the robot cannot go to the back to execute Recognition and crawling. In RGB red extraction, when the red element is much larger than the green and blue elements, that is, the red RGB value minus the green and blue values are greater than the threshold  $diff\_R$ , we think the object is red. According to the definition of computer color, we set the domain value  $diff\_R=100$ . After extracting the color range corresponding to the image, we record the coordinate range corresponding to the color graphic, and use the smallest circumscribed quadrilateral to display the range of the object. By calculating the coordinates of the camera coordinate system relative to the world coordinate system, we can substitute it into the world coordinate system [3]. The original scene, target color and approximate range recognition results are shown in the figure below.

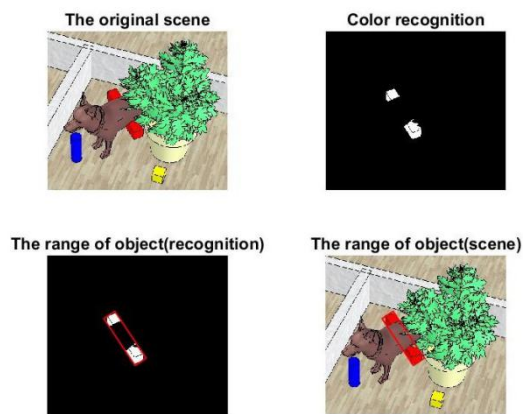


Figure 3. Recognition results of the target object

### 4 Sub-pixel Harris corner extraction

Through recognition and operation, we can determine the general range of the object. However, the presence of other objects will greatly limit the operation range of the robotic arm, and the target object is mostly obscured by other objects. Therefore, we need to further narrow the range of the object and find the suitable grasp surface of the target object. In order to mark the precise position of the object and facilitate the marking of the coordinates of the grasping surface, we have introduced corner detection.

### 4.1. Harris corner detection

Harris operator is an operator based on the improved feature point extraction of Moravec algorithm. It can be substituted into any direction through Taylor expansion and deformation to calculate the pixel value change in any direction [4].

For the local translation, the autocorrelation function of the image is used to represent the change in the gray value in the window, and the Taylor series expansion is used to derive the autocorrelation function as:

$$E(\Delta x, \Delta y) \approx \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} M \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad (1)$$

$$\text{and } M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} \quad \begin{aligned} A &= \sum_w (I_x(x_i, y_i))^2 \\ B &= \sum_w (I_y(x_i, y_i))^2 \\ C &= \sum_w I_x(x_i, y_i) I_y(x_i, y_i) \end{aligned} \quad (2)$$

W is the Gaussian window.

The corner response function is calculated as follows:

$$CRF = \det(M) - k(\text{trace}(M))^2 \quad (3)$$

Among them, k is the empirical value, generally 0.04-0.06. detM is the product of the eigenvalues of the matrix M, and traceM is the sum of the eigenvalues of the matrix M.

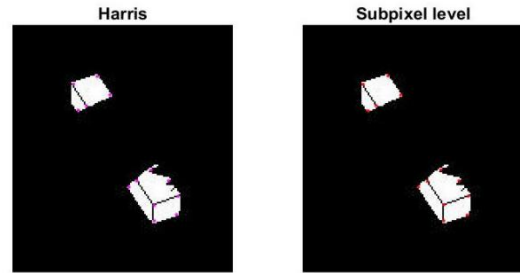
When the corner response function values of all positions in a picture are calculated, we set the threshold T and find the local maximum. When a certain point simultaneously meets the local maximum value and the corner response function value is greater than the threshold T, we consider the point to be a corner point. Harris corner detection is relatively simple and easy to implement, but it is less suitable for small object detection. For small targets in complex situations, we need to increase the accuracy of corner detection. As a result, we accurate the corner coordinates to the sub-pixel level, and compare with the effect.

### 4.2. Harris corner detection at sub-pixel level

In order to obtain sub-pixel-level corner coordinates, we introduce an iterative algorithm for optimization [5]. For the identified points near the corner points in Harris detection, the gray gradients are all perpendicular to the line connecting the point and the corner point. The actual image will be affected by noise, and corresponding errors will occur. In order to minimize the error sum, we can use an iterative algorithm to optimize the solution, and sum the results, then the corner points can be optimized at the sub-pixel level.

### 4.3. Corner recognition results

We perform Harris and sub-pixel corner detection on the recognized images, and the results are as follows.



**Figure 4.** Harris and sub-pixel corner detection results

At the same time, we get the specific coordinates of the corner points obtained by the two detections through matlab, as shown in the following table.

**Table 1.** Comparison of Harris and sub-pixel-level corner detection specific coordinates

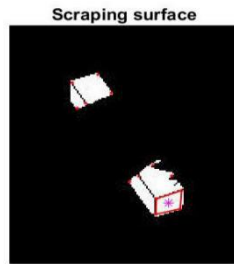
Harris(x)	Harris(y)	Subpixel(x)	Subpixel(y)
39.0000	39.0000	38.6883	39.1741
42.0000	56.0000	41.7658	56.4178
47.0000	53.0000	46.7874	52.9504
54.0000	34.0000	54.3356	33.6151
62.0000	46.0000	62.4107	46.1158
74.0000	106.0000	73.5755	105.9301
79.0000	102.0000	79.2108	102.2966
88.0000	95.0000	87.7649	94.8183
90.0000	118.0000	89.7627	117.5075
90.0000	128.0000	127.9808	89.5163
99.0000	102.0000	99.1373	102.0710
104.0000	124.0000	104.2867	124.2542
105.0000	112.0000	105.1402	111.8384

In conclusion, it can be seen that the corner coordinates obtained through sub-pixel level detection are more accurate and can reflect the exact position of the object itself.

## 5 Confirm the grab point

According to the corner coordinates obtained by the corner detection, we can get the more accurate range of the target object. Since the RGBD camera can measure the three-dimensional coordinates at the same time, we can substitute it into the world coordinate system[3].

In the robot arm object grasping, we need to make the coordinate system of the hand grasp coincide with the center of gravity coordinate system of the object [3]. However, in this experimental state, the center of gravity of the target object is obscured, and the manipulator cannot grasp normally. In order to facilitate the robot's grasping, we need to find the object surface with less occlusion. From this, we display the grabbing surfaces with small or no gaps in the corner point range, and calculate the approximate grab points through the corner points, as shown in the figure below.



**Figure 5.** Target grab surface and grab point

Where \* is used to display the location of the grab point.

## 6 Conclusion

This paper introduces a method of visual recognition and image processing of targets in complex conditions, which is convenient for robots to perform corresponding operations on targets in complex conditions. Through V-REP and MATLAB joint simulation, for the target object in complex state, the robot uses RGBD camera to input images and XYZ coordinates. The RGB value determines the target range, and introduces sub-pixel Harris corner detection to establish the grasping surface and center point coordinates for feedback to the robot. This method enables the robot to find a more suitable grasping point for the target object in complex state, so that the robot can operate the target object more effectively.

## References

1. YE Hui, ZHANG Weimin. Development of Intelligent Robot Grasping System Based on Machine Vision. *J. Science & Technology Vision*, 12: 01-05 (2016)
2. ZHANG Shichao, TAO Cunbin. Application of Robot Simulation Based on V-REP and MATLAB. *J. SHIPBOARD ELECTRONIC COUNTERMEASURE*. 43: 111-114, 128 (2020)
3. LI Lei, WU Yang, QIU Xudong. Visual Positioning Method for Handle of Aircraft D. 3oor. *J. Science & Technology Vision*. 17: 31-34 (2017)
4. CAI xinzhan, LIU yanyan. Fast Image Registration Algorithm Based on Adaptive Screening Harris Corner Detection. *J. Semiconductor Optoelectronics*. 41: 758-761 (2020)
5. YAO Yao, HU Zhong-xiang, SHI Xiao-jun. Camera self-calibration on a sub-pixel level to apply extract corners by Harris algorithm. *J. Electronic Design Engineering*. 17: 61-62, 65 (2009)