Real-Time Multi-object Grasp Based on Convolutional Neural Network

Hui Yu and Yulin Xu
School of Mechatronics Engineering and Automation, Shanghai University, Shanghai 200444, China
Email: shangdahuihui@163.com

Abstract. The grasping of the robotic arm is the basic ability of the robot to perform a variety of complex tasks. For the robotic arm, visual-based grasping tasks are still a challenging problem. This paper presents a system to implement dynamic grasping of the robot arm. The neural network is used to detect objects in multiple scenes, and then the robot arm is controlled to achieve closed-loop grasp of the target object. In the process of grasping the target object, if the position of the target object moves, the robot arm will immediately stop and go to the new target position for grasping. The extensive experiments demonstrate that the proposed method enables the robotic arm to achieve dynamic grasping in real world.

1. Introduction
In recent years, owing to the effective work of the robot arm, it has been widely used in production and life. Among them, vision-based robot arm grasping has gradually become a current research hotspot. The traditional robotic arm usually performs the target grasping work according to the established process, and cannot receive external information. When the target position changes, it is necessary to reset the control program. The machine can obtain the information of the external environment effectively by computer vision.

In 2006, the Hinton et al. first proposed the concept of deep learning. Compared with the traditional method of manually extracting features, the advantage of deep learning is that the feature extraction process does not require the user to extract any features in advance, but uses a general learning process to enable the model to learn from large-scale data and then learn to have the goal characteristics [1-2].

Deep learning also plays a big role in grasping objects with robotic arm. Ref. [3] uses the grasp function to make score predictions for all possible target positions. The smooth grasp function makes the model robust to uncertain grasp positions, and the grasp function is obtained by convolutional neural network. Ref. [4] proposed a two-step cascade detection system, by processing RGB-D multi-modal information to detect the optimal grasping position of the target object, and obtained a good grasping effect.

Most objects in real life can be grasped in different ways, and there would be an optimal way in a local region if it exists. In most cases, the optimal grasp is preferred to be executed, humans grasp objects easily with the advanced visual feedback. It is essential to provide an accurate grasp pose for robot arm to grasp object, so it is still a difficult problem to grasp the target object for the robot arm at the moment.

This paper aims at the limitation of traditional robotic arm that cannot be autonomously grasped. Taking Kinect v2 vision sensor and Jaco arm as the re-search object, the machine vision technology is
applied to the robotic arm grasping task to improve the perception ability of the robotic arm grasping system and realize the autonomous robotic grasping function.

2. System Platform Construction
The robotic grasping system in this paper is composed of visual sensors, mechanical arm system and main control computer. The grasping experiment is based on the robot experiment platform, and the relevant function modules are designed on the ROS system. The robotic arm system mainly consists of Jaco arm and chassis. The construction model and process of the whole system are shown in figure 1.

2.1. System Hardware Design
(1) Kinect v2: adopt Microsoft Company Kinect v2 camera, the maximum camera resolution is 1920 × 1080, which meets the system design requirements.
(2) Computer: a computer equipped with Intel I7-8700K processor, Nvidia GTX-2070 graphics card, responsible for receiving images collected by Kinect camera. After completing the object recognition and positioning through algorithm processing, the three-dimensional coordinate information of the object is sent to the robot arm and the planned movement of the robot arm is controlled.
(3) Jaco arm: Jaco arm is a flexible, lightweight, and easy-to-operate bionic robotic arm developed by Kinova Company of Canada. It has 3 degrees of freedom at the end gripper. The Jaco arm has a long arm span, because each joint integrates a high torque brushless DC motor and a Harmonic Drive reducer, which makes its structure very compact. Jaco arm supports the development under the ROS, the selection of Jaco arm can well meet the needs of this experiment.

2.2. Calibration of Kinect V2
For the robotic grasping system, in order to improve the accuracy of object grasping, the primary work is to accurately identify and locate the object. The first step of target recognition and positioning is to calibrate the Kinect v2 vision sensor to obtain the internal parameters of the camera, and to obtain the external parameters describing the sensor and the world coordinate system by calculation. Finally, the target object’s 3D coordinate value in the manipulator coordinate system is obtained by coordinate transformation based on the depth image extracted by Kinect v2 [5].

Before the experiment, the pixel coordinate \((u, v)\) of the center of the object grasp frame should be calculated in the camera coordinate system \((x_r, y_r, z_r)\). The relationship between coordinate systems is converted to:
\[
\begin{bmatrix}
  x_r \\
  y_r \\
  z_r
\end{bmatrix} = d I_m^{-1} \begin{bmatrix}
  u \\
  v \\
  1
\end{bmatrix},
I_m = \begin{bmatrix}
  \beta_x & 0 & u_0 \\
  0 & \beta_y & v_0 \\
  0 & 0 & 1
\end{bmatrix}
\]

\(I_m\) is the internal parameter of the camera, \(d\) is the depth value corresponding to the coordinate point \((u, v)\). Secondly, the coordinates \((x_r, y_r, z_r)\) in the camera coordinate system are calculated in the corresponding coordinates \((x_b, y_b, z_b)\) in the manipulator base coordinate system. The relationship between coordinate systems is converted to:

\[
\begin{bmatrix}
  x_b \\
  y_b \\
  z_b
\end{bmatrix} = \begin{bmatrix}
  R & T \\
  0 & 1
\end{bmatrix} \begin{bmatrix}
  x_r \\
  y_r \\
  z_r
\end{bmatrix}
\]

\((x_b, y_b, z_b)\) is the coordinates of the target object in the base coordinate system of the robot arm. The schematic diagram between each coordinate system is shown in figure 2.

**3. Object Detection**

The target recognition algorithm is a crucial part of the robotic arm grasping system. The accuracy of the recognition directly affects the success rate of the subsequent grasping work. Through research and analysis of the current status of target detection technology, this paper selects the YOLOv3 algorithm as object detection algorithm, which is based on regression method to identify and locate the target object in real time.

**3.1. YOLOv3 Network Model**

YOLOV3 is composed of convolution layers, leaky relu and batch normalization layers. The network basically adopts full convolution, while introducing residual block structure, which reduces the training difficulty of deep network and greatly improves the accuracy and speed of target object detection [6-7].
The backbone of YOLOV3 is DarkNet-53, and the input image is first feature extracted. In order to obtain fine-grained features, Feature Pyramid Networks (FPN) of YOLOv3 [8] use convolution with step size of 2 to downsample, carries out target detection at 32, 16, 8 times downsampling, splices the feature information obtained by three downsamples through the passthrough layer, and uses three different-scale feature map fusions (13×13, 26×26, 52×52) in the multiscale detection section to connect shallow feature maps to deep feature maps, the fusion of features of different scales enables YOLOV3 to learn deep and shallow feature information. The YOLOV3 network structure is shown in the following figure 3.

Figure 3. Structure diagram of YOLOv3.

YOLOv3 uses regression to extract features, it is an end-to-end training process. After inputting the image, the category and bounding boxes of the object on the image are directly returned in the feature layer. Compared with the two-step method of first detection and then regression such as faster R-CNN, this one-step method significantly improves the real-time performance. YOLOv3 has the following characteristics:

(1) Darknet-53 network architecture

In the process of feature detection on images, YOLOv3 uses Darknet-53 network to replace YOLOv2’s Darknet-19 network. YOLOv3 draws on the method of Residual Network (Residual Network) to set up shortcut links at certain layers, thereby improving efficiency.

(2) Prediction of bounding box

Yolo3 uses the multi-scales strategy, there are 3 branch outputs for prediction. The output feature map sizes are 13*13, 26*26, 52*52. Each feature map uses 3 anchor boxes, the anchor boxes of YOLOv3 are obtained by clustering method. The parameters of the target bounding box are calculated as shown in figure 4.

Figure 4. YOLOv3 bounding box calculation.
(3) Multi-scale prediction

YOLOv3 uses multi-scale feature map fusion to make prediction, and uses the fusion (fusing 3 layers) and upsampling methods in the FPN network to perform multi-sample and multi-size detection. At the 82nd, 94th and 106th layers of the network, multi-scale prediction is carried out, which greatly improved the accuracy of YOLOv3 detection of small target objects [9-10].

4. Grasp Experiment

Before the grasping action is executed, the robotic arm must be provided with 3D position information of the target object's grasping point in the robotic arm's base coordinate system, and then the robotic arm can successfully grasp the target object. In this paper, select the center of the rectangular frame of the target object as the grasping point.

The image acquisition device used in this experiment is Kinect v2. Using Kinect v2, the depth information corresponding to the center of gravity of the grasp frame can be obtained. Through the coordinate relationship between the camera and the robot arm, the 2D coordinate point of the center of gravity of the grasp frame and the depth information corresponding to the coordinate point can be converted into the 3D position information of the actual grasp point in the base coordinate system of the robot arm. After successfully completing the calibration of Kinect v2 in section 2.2, start the grasping experiment of the target object.

4.1. Single Object Recognition Grasp Experiment

Choose an apple and place it on the experimental table at random. After building the experimental platform and successfully calibrating Kinect v2, enter the command to grasp the apple and place it in the designated storage basket in the system, the robotic grasp system immediately executes the grasping and placing task. The process decomposition diagram of the system to identify and grasp apple is shown in figure 5. As shown in figure 5a, the robot arm starts to move. Figure 5d shows that the robot arm successfully grasps the apple in the 7th second. Figure 5h, the robot arm puts the apple in the storage basket in the 13th second.

![Figure 5. Single object grasp experiment.](image-url)
4.2. Object Grasp Experiment in Complex Environment

Choose different items and place them on the experimental table at random, so that the objects are not next to each other. Because the robot arm has a certain size, if it collides with other fruits in the middle, it will be regarded as a failure to grasp.

After setting up the experimental platform for this scenario and successfully calibrating Kinect v2, first select the apple on the desktop as the target grasping object, enter the command to grasp the apple and place it in the designated basket in the system, and the robotic arm immediately executes the task of grasping the apple. After successfully grasping the apple and placing them in the designated basket, then enter the command to grasp the orange in the designated basket in the system, and the robotic arm grasping system immediately executes the task of grasping orange.

In the process of grasping, the Kinect v2 image data is read in real time. It can be seen that the YOLOv3 algorithm accurately recognizes apple, orange, banana and bottled water on the desktop. Even during the grasping process, the object detection algorithm can still detect the object captured by the robot arm and the object to be shielded, the real-time requirement of the target detection algorithm for this experiment is further validated. The process decomposition diagram of the robot grasp system to grasp the apple first and then the orange is shown in figure 6, the following screenshot shows the continuous grasp process of the manipulator.

Through experimental verification, it can be seen that the robotic arm grasping system designed in this paper can not only meet the object recognition and grasping in a single scene, but also meet the accurate recognition and real-time grasping of target objects in complex scenes.

Figure 6. Continuous grasp full-process video screenshot in complex scenes.
5. Conclusion
A robot arm grasping system which can realize the dynamic grasping of the robot arm in multi-object scene is built in this paper. The experimental results of grasping on jaco arm show that the method in this system can accurately identify the target objects, which demonstrate the effectiveness of the target detection method in practical application scenarios.

The goal of future research is to use a model to realize the detection, classification and grasping position detection of the target object, which can more effectively combine the two network models into a whole, and finally perform the target object detection and the grasping position detection in the meantime.

References
[1] Krizhevsky A, Sutskever I and Hinton G E 2012 ImageNet classification with deep convolution neural networks Advances in Neural Information Processing Systems (Cambridge, USA: MIT Press) pp 1097-1105.
[2] LeCun Y, Bengio Y and Hinton G 2015 Deep learning Nature 521 (7553) 436-444.
[3] Johns E, Leutenegger S and Davison A J 2016 Deep learning a grasp function for grasping under gripper pose uncertainty IEEE/RSJ International Conference on Intelligent Robots and Systems (Piscataway, USA: IEEE) pp 4461-4468.
[4] Lenz I, Lee H and Saxena A 2013 Deep learning for detecting robotic grasps International Journal of Robotics Research 34 (4/5) 705-724.
[5] Wiedemann L G, Planinc R, Nemec I, et al. 2016. Performance evaluation of joint angles obtained by the kinect V2 Iet International Conference on Technologies for Active & Assisted Living.
[6] Redmon J, Divvala S, Girshick R, et al. 2015 You only look once: Unified, real-time object detection Proceedings of CVPR pp 779-788.
[7] He K, Zhang X, Ren S, et al. 2016 Deep residual learning for image recognition IEEE Conference on Computer Vision & Pattern Recognition (IEEE Computer Society).
[8] Lin T-Y, Dollar P, Girshick R, et al. 2017 Feature pyramid networks for object detection 2017 IEEE Conference on Computer Vision and Pattern Recognition pp 936-944.
[9] Huang G, Liu Z, Van Der Maaten L and Weinberger K Q 2017 Densely connected convolutional networks Proc IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. pp 4700-4708.
[10] Redmon J and Farhadi A 2017 YOLO9000: Better, faster, stronger Proceedings of the Computer Vision and Pattern Recognition pp 6517-6525.