Construction of Virtual Datasets for Bin Picking

Zhuang YANG\textsuperscript{1}, Jian-jun YI\textsuperscript{1,}\textsuperscript{*}, Liang HE\textsuperscript{2,3}, Ya-jun ZHANG\textsuperscript{1} and Bo ZHOU\textsuperscript{1}

\textsuperscript{1}Department of Mechanical Engineering, East China University of Science and Technology, Shanghai 200237, China
\textsuperscript{2}Shanghai Aerospace Control Technology Institute, Shanghai 201109, P.R. China
\textsuperscript{3}Shanghai Key Laboratory of Aerospace Intelligent Control Technology, Shanghai 201109, P.R. China

*Corresponding author

Keywords: Bin Picking, Virtual Dataset, Virtual Simulation.

Abstract. Bin Picking is a typical problem for identifying and sorting complex stacked objects, the difficulty of which lies in the identification of objects. This paper uses deep learning method to identify objects preliminarily. Because the deep learning method requires high-quality datasets, the traditional manual annotation method is too difficult to complete under the industrial production environment. Therefore, this paper proposes a method to build virtual datasets based on virtual simulation technology. It uses OpenGL technology and Bullet physics engine to build virtual scenes. Then through camera loading and scene drawing, the datasets including RGB images, segmentation annotation images and so on are generated. Experimental results show that the model trained on virtual datasets by deep learning can achieve good accuracy, so the feasibility and practical application value of the method are verified.

Introduction

Nowadays, industrial robots have been widely used in assembly, sorting, handling, palletizing and other industrial environments, greatly improving the efficiency of production. However, for the recognition and sorting of complex stacked objects, most of them are still completed manually. Therefore, it is of great significance to realize the automation of industrial robot processing tasks, and the focus of which is to solve the problems of automatic recognition and grasping objects.

Bin Picking is a classical problem of this kind, and it is also a common process in various industrial processes. In short, it is to use a mechanical arm to take out the disordered objects in the box. It is very simple to do it by hand, but it is very challenging to do it by robot. In recent years, with the progress of academic circles such as computer vision and motion planning, the Bin Picking problem began to have stable and reliable solutions and the possibility of commercialization [1].

In Bin Picking, the current solution is basically model-based method. This method calculates the type and three-dimensional pose of each object through RGB images or point cloud, projects the grasping points of the object into the world coordinate system, and selects optimal grasping points according to other requirements (collision detection, etc.) [2].

Based on the model-based method, we use the deep learning method to classify and segment the RGB image, and combine the depth image to assist point cloud segmentation, so as to determine the object’s pose and grasping points. The main process is shown in Fig. 1 below. In this paper, we focus on the virtual dataset generation part.

With the method of deep learning, the quality of datasets has a direct impact on the learning effect. However, the datasets used in practice are difficult to meet the requirements of diversity, and they take a lot of time and manpower to take pictures and label pictures, but the quality of datasets cannot be guaranteed. Because manual labeling is a very subjective process, different people may have different results in the labeling of the same picture, and errors may occur in the labeling process. These subjective factors will affect the quality of deep learning [3].
Existing datasets now include PASCAL VOC, ImageNet, MS COCO and so on. These datasets, such as PASCAL VOC and ImageNet, lack segmentations and labels of objects in the images [4,5]. Although MS COCO has a large number of segmented annotation datasets, the categories are mainly concentrated on some common classes, such as aircrafts, bicycles, cars, cats, dogs, tables, chairs and so on, which are difficult to meet the needs of industrial production [6]. In this paper, the categories of objects are mainly concentrated on metal workpieces, such as keys, screws, oil pans, angle brackets and so on. Therefore, the existing datasets are difficult to meet the requirements of this topic.

In order to produce datasets to meet the need of Bin Picking and avoid using a large number of manpower, this paper uses virtual simulation technology to generate virtual datasets. Using virtual simulation technology can generate a large amount of data in the field of metal workpiece, label automatically, avoid wasting a lot of manpower, and guarantee the quality of labeling.

Using virtual simulation technology to make virtual datasets, Yonglin Tian et al. [7] use City Engine and Unity 3D technology to build virtual scenes of cities, streets and vehicles, and use C# script to make objects such as vehicles move, and use three-angle cameras to take pictures to generate the virtual datasets of vehicles.

This paper adopts the virtual simulation technology based on OpenGL and Bullet, mainly because it can customize the objects and backgrounds freely. Also, the computational complexity is small, so it can generate a large number of virtual datasets quickly.

Construction Steps of Virtual Datasets

OpenGL Scene Construction

To build a virtual dataset, the first step is to build a 3d simulation scene in OpenGL. According to the requirements of the subject, it mainly focuses on metal parts, including keys, screws, oil pans, angle brackets and so on. For the components, SolidWorks or other 3d CAD software can be used for 3D modeling. 3DS MAX is used to apply the map through UV mapping to fit the appearance of the actual object and lighting model parameters, and then the model file of the corresponding format is exported. Finally, the Assimp model loading library is used to import the model into the OpenGL simulation 3d scene.

Using the Bullet Physics Engine to Obtain the Stacked Scene

After using the Assimp library to load the model files to the OpenGL 3d simulation scene, it only loads the models to preset fixed positions in the scene, and cannot imitate the gravity, collision,
deformation and other factors in the real world. Bullet engine is an open source physical simulation computing engine that allows us to simulate realistic scenarios.

In the OpenGL scene, we built a container and use the Bullet physics engine to create a virtual scene that conforms to the physical motion rules in this container. After initializing the physical engine, multiple models corresponding to the dataset are loaded into it, which fall freely under the action of gravity and collide with each other until they are completely stationary, forming a stacked scene at the bottom of the Bullet container. However, when there are a large number of models in the scene, the simulation process of collision motion often takes a long time or even gets stuck, because the object is doing subtle motion for quite a long time before it is completely stationary. This affects the efficiency of dataset generation, but has little impact on the quality of dataset generation. Therefore, when building the scene, after passing a certain time threshold, we stop the physical calculation of Bullet and draw the scene directly.

In the process of drawing such a scenario, each model is assigned a segmentation ID, which is used to mark each individual model and prepare for subsequent steps.

**Loading the Camera of the Virtual Scene and Acquiring the RGB Images**

In order to obtain the image of the stacked scene as the dataset, the camera needs to be placed in a certain position of the virtual 3D simulation scene to shoot the stacked scene. To obtain the RGB image, the scene rendered in the OpenGL viewport needs to be saved as an image. To realize this process, useful information of the buffer in OpenGL needs to be extracted. GlReadPixels function and Freemage image processing library are used here. GlReadPixels can be used to read the color sequence in the image cache, and then the color sequence can be saved into the RGB image through the help of Freemage library.

**Acquiring the Segmentation Annotation Images**

The important point of acquiring virtual datasets is to automatically complete the annotation and avoid using a lot of manual work. In essence, the process of automatic labeling is to obtain the segmentation labeling images of objects. In the segmentation annotation diagram, the same object has the same gray value, while different objects have different gray values, so that each object can be clearly distinguished from the gray level, as shown in Fig. 2. Because the gray value of the segmentation annotation diagram is small and cannot be seen directly, the library of matplotlib is used here to visualize the segmentation annotation diagram. In the diagram, different colors represent different gray value.

Compared with manual annotation, the annotation of virtual dataset is more accurate, especially the arc around the object can be accurately marked, while traditional manual annotation can only manually use inaccurate polygon annotation to fit the arc, as shown in the Fig. 2. Another point is that the annotation of virtual dataset can clearly mark the holes in the object contours, while the traditional manual annotation can only mark the object contours. A large number of metal parts are porous, and the structure of the hole is also important for the object grasp. At this point, the automatic annotation of virtual datasets is more consistent with the requirements of Bin Picking than the manual annotation.

![Figure 2. Visual segmentations of objects.](image-url)
Obtaining segmentation annotation images, namely, the process of automatically completing annotation is similar to obtaining RGB images. It is just to render the color of each model in the scene into a gray value equal to its segmentation ID value, and then read the image of the stacked scene in the viewport.

Recording the Categories of Objects with YAML Files

In the above processes, we get the RGB images and segmentation annotation images of the objects in Bullet container, but the category information of the objects is missing. If the virtual dataset lacks the information, then the deep learning model trained by the dataset cannot learn the knowledge of classification. Therefore, it is very important to record the category information of objects. From the previous steps, we know that in the RGB image and the corresponding segmentation annotation image, each object has a segmentation ID value, which is equal to the gray value of the corresponding object in the segmentation annotation image. In the virtual dataset, YAML files record the segmentation ID value of the object and the corresponding category, and by segmentation ID value, the object with the same gray value can be found, thus adding the category information of the object in the dataset.

Experiments

In the experiment, the flat key model, the angle bracket 4040 model, the angle bracket 6060 model and the mixed angle bracket model were respectively used to construct the virtual datasets, which included RGB images, segmentation annotation images and YAML files. As shown in the left image of Fig. 3, they are the RGB diagrams of each model in the virtual dataset. The segmentation annotation diagrams of the virtual datasets are shown in right image of Fig. 3.

In order to verify the feasibility of using virtual dataset instead of real dataset in deep learning training, we built a real scene for taking pictures, including industrial camera, industrial lens, light source, image acquisition card, brackets and purchased relevant parts. The final real scene built is shown in Fig. 4.
Using this real scene, we took real pictures of the flat key, angle bracket 4040, angle bracket 6060 and the mixed angle bracket. Mask R-CNN is the latest achievement in the field of deep learning and computer vision. It can complete target detection, target classification and pixel-level target segmentation with high accuracy [8]. We used the model trained by the virtual dataset to do the test on these real images, as shown in Fig. 5.

In the experiment, about 20 real images were taken for each model, and then the deep learning model obtained through training on the virtual dataset was used to do the inference of these real images. The obtained segmentation and classification accuracy data is shown in Table 1. Among them, the object segmentation precision IOU refers to the ratio of the intersection of the detection result and Ground Truth to their union. The precision rate of the object measures the percentage of the results which are correct, while the recall rate measures the ratio of the correctly predicted examples to all positive examples in the sample. It can be seen from the table that the model trained by the
virtual dataset has a better effect on the real picture. Therefore, it is feasible to use the virtual dataset instead of the real dataset to train the deep learning model in Bin Picking. The problem of vision is solved, and the inference results of deep learning model can assist the segmentation of point clouds and the estimation of poses, so as to complete the grab planning, motion planning, and complete the work of Bin Picking.

Table 1. Classification and segmentation results of different models.

| Model               | IOU(%) | Precision(%) | Recall(%) |
|---------------------|--------|--------------|-----------|
| flat key            | 86.2   | 99.0         | 91.0      |
| angle bracket 4040  | 84.6   | 98.2         | 94.9      |
| angle bracket 6060  | 91.5   | 98.5         | 96.0      |
| mixed angle bracket | 83.7   | 92.8         | 96.2      |

Summary

Aiming at the problem that the disorderly placement of objects in Bin Picking material box leads to the difficulty of identification, this paper uses the deep learning method for instance segmentation. Since deep learning methods require high-quality datasets, which are traditionally done by real shooting and manual labeling, it is difficult to do so in an industrial production environment. Therefore, this paper puts forward the idea of constructing virtual datasets through virtual simulation technology, gives a concrete construction method, and verifies the feasibility of this method through experiments.

Acknowledgement

This paper was supported by the Military and civilian integration project (Shanghai economic and information commission) under Grant No.201720, the Natural Science Fund of China (NSFC) under Grant Nos. 51575186, 51275173, and 50975088, Shanghai Software and IC industry Development Special Fund under Grant No. 180121, and Shanghai Science and Technology Action Plan under Grant No. 18DZ1204000, 18510745500.

References

[1] Schyja A, Kuhlenkötter B. Virtual Bin Picking—a generic framework to overcome the Bin Picking complexity by the use of a virtual environment[C]//Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH), 2014 International Conference on. IEEE, 2014: 133-140.

[2] Kehl W, Manhardt F, Tombari F, et al. SSD-6D: Making RGB-based 3D detection and 6D pose estimation great again[C]//Proceedings of the International Conference on Computer Vision (ICCV 2017), Venice, Italy. 2017: 22-29.

[3] Ros G, Sellart L, Materzynska J, et al. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 3234-3243.

[4] Everingham M, Van Gool L, Williams C K I, et al. The pascal visual object classes (voc) challenge[J]. International journal of computer vision, 2010, 88(2): 303-338.

[5] Deng J, Dong W, Socher R, et al. ImageNet: A large-scale hierarchical image database[J]. Proc of IEEE Computer Vision & Pattern Recognition, 2009:248-255.

[6] Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//European conference on computer vision. Springer, Cham, 2014: 740-755.
[7] Tian Y, Li X, Wang K, et al. Training and testing object detectors with virtual images[J]. IEEE/CAA Journal of Automatica Sinica, 2018, 5(2):539-546.

[8] He K, Gkioxari G, Dollár P, et al. Mask R-CNN[C]//2017 IEEE International Conference on Computer Vision (ICCV). 2017.