# Garbage sorting simulation of Panda robot arm based on deep learning and visual recognition

Xinzi Liu, Haoran Wang, Yutao Guo, Sijing Qu

***Abstract*—This experimental study presents a comprehensive approach to garbage sorting using computer vision and robotic manipulation. By integrating a composite camera into the pybullet simulation environment, images of the target region are captured and divided into distinct regions, each containing a single piece of garbage. Leveraging an internal package, precise object coordinates relative to the camera system are obtained.**

**Efficient garbage classification is achieved through training a ResNet convolutional neural network on a dedicated garbage model dataset. The model exhibits remarkable accuracy, correctly classifying single garbage items into four categories: recyclable garbage, other garbage, kitchen waste, and hazardous garbage, with an impressive classification accuracy exceeding 97%. Additionally, the model demonstrates high accuracy in distinguishing among 216 subcategories of objects.**

**To enable the robot arm to act on the identified garbage, relative camera coordinates are transformed into absolute coordinates within the arm's coordinate system. These coordinates, along with the identified garbage types, serve as input parameters for the panda robot arm's simulation, enabling it to execute the necessary sorting actions. Consequently, the garbage in the target area is effectively sorted and directed to their designated areas based on their respective classifications.**

**This integrated approach offers a promising solution for automated garbage sorting, combining computer vision, machine learning, and robotic manipulation. The achieved results highlight its potential for real-world applications, contributing to waste management efforts and promoting sustainable practices.**

## I. Introduction

The increasing standard of living worldwide has led to a substantial rise in daily household waste generation. According to the World Bank, global waste is projected to surge by 70% to 3.4 billion tons by 2050, posing a significant challenge in terms of effective waste management. Garbage classification serves as a fundamental prerequisite for implementing scientific waste treatment practices, facilitating the recycling of recyclable materials and the safe disposal of hazardous waste. With the introduction of Household Garbage Classification Systems and successful trials in various cities, garbage classification has gradually become an integral part of people's daily lives.

Garbage classification offers multiple benefits: it helps minimize resource wastage and reduces environmental pollution, contributing to the establishment of an ecological civilization. However, the current implementation of garbage classification heavily relies on active participation from residents and supervision by authorities, which places certain demands on residents' awareness and requires significant manpower. Despite these efforts, the overall impact has been limited. In recent years, the rapid advancements in machine vision technology have provided a promising solution by utilizing machine vision algorithms for garbage classification.

This research paper proposes a garbage sorting system based on pre-trained deep learning neural networks, comprising an algorithm module and a simulation module. The algorithm module encompasses functions such as data augmentation, model pre-training, model fine-tuning, and model inference. The features of the photos are extracted using the ResNet50 network model with the aid of the Adam optimizer. The Huawei Cloud Artificial Intelligence Competition dataset is employed for model training to verify its accuracy and meet the experimental requirements. Subsequently, the system is deployed within the pybullet simulation environment, and a garbage module is developed to simulate and complete the development of the garbage classification system.

## II. Related Work

In recent years, there has been a significant increase in research focusing on the application of deep learning in the field of artificial intelligence. Researchers have been training neural networks to obtain various evaluation models and achieve the classification of diverse objects, future model predictions, and visual perception. Our understanding of intelligent garbage classification primarily relies on the study of image recognition neural networks through observation. These trained networks can effectively classify objects. However, for the network itself, it cannot accomplish the task of target recognition by solely observing object features. Instead, it requires the conversion of images into binary data and the transformation of graphical data into matrix-formatted data that can be input into the network for learning.

Existing environments such as ROS and Pybullet have gained widespread usage in robot simulation. Considering the impracticality of physically testing with a robot arm, we adopt the packaged panda arm, which can be readily imported into Pybullet, simplifying the process of simulating our robot arm.

Over the past five to ten years, numerous researchers have combined deep learning, vision, and mechanical arms to manipulate objects. For instance, Zhihong et al[1]. employed a similar approach to ours in sorting garbage, while Guanhao et al[2]. utilized the YOLOv5 system for garbage recognition and classification. Additionally, Zhifei et al[3]. proposed a solid waste sorting system based on industrial robots.

## III. Data

This project uses data sets from Baidu Flying Paddle to train the classification network, a total of 56528 JPEG type samples,

including kitchen waste, harmful garbage, recycling and other four categories, And each subdivision subclass a total of 214 species. The network training uses a collection of photos extracted from the data to train through 80 epochs. The robot simulation is done using PyBullet. PyBullet is a packaged python module that supports loading URDF, SDF, MJCF and other robot description files, and provides forward/reverse kinematics, collision detection and other functions. In this project, the panda arm model built in PyBullet was used, and the distance between jaws was increased to adapt to the size of the model. Model using open source URDF and OBJ files, source link https://github.com/ChenEating716/pybullet-URDF-models. We scaled the model used according to the project content and the size of the robot arm, and modified the origin position. In order to verify the accuracy of the input image in the simulation environment, tests were conducted 30/40/50 times and 10/15/20/25 times respectively with two standards. The accuracy of classification network was verified by using the correct simulation visual output as input. 100 epochs were iterated and four of them were selected to obtain the accuracy of four major categories and 214 minor categories. Finally, in the simulation grasping part, in order to verify the effectiveness of grasping, we divided objects into four categories according to the shape, fixed grasping distance, and tested each category of objects 50 times. Then fix it as a square object, set four grasping distances, and test 50 times in each group.

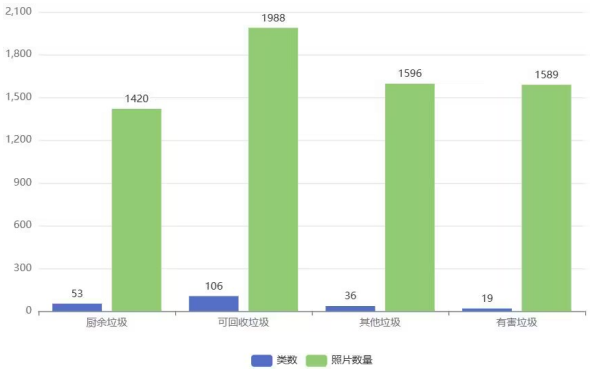


Fig. 1. The amount of each type of garbage and the total number of photos

## IV. Methods

### A. Panda Robot Simulation

In this project, we use pybullet for simulation. Pybullet is a physics engine and robotics simulation library. It provides a Python interface to the Bullet Physics Engine, allowing users to simulate and control robots, perform physics-based simulations, and create interactive virtual environments for various applications such as robotics, computer graphics, and machine learning.

In the first step, we initialize the pybullet simulation environment, such as building the basic coordinate axis (the y-axis is vertical), setting the gravity, setting the time interval, and so on. In this project, we refer to the robot model that comes with the bullet library, so we need to call the panda robot to enter the simulation environment. Then, directly above the coordinate plane, we import a camera point, which is directly on the xz plane. After defining the boundaries of the relevant simulation environment, we import the relevant items and trash box models, and randomly distribute them within a certain range, while ensuring that they are within the visual range of the camera.

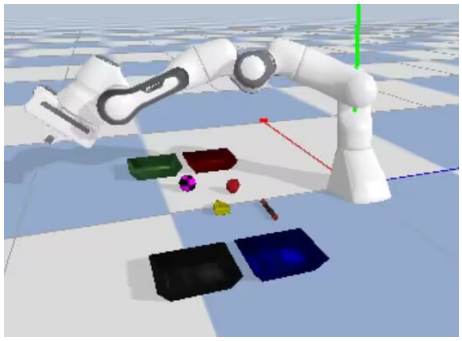

Fig. 2. The maximum working range of the model

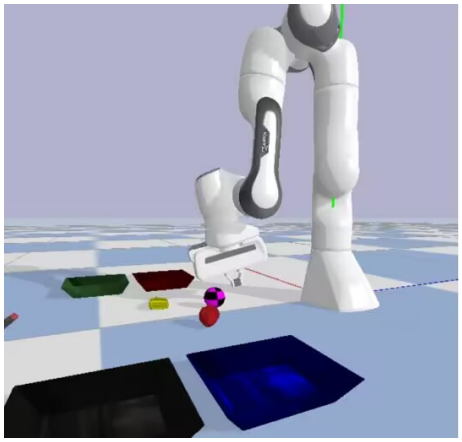

Fig. 3. The minimum working range of the model

After importing the item model, we need to initialize the panda robotic arm. The Panda robotic arm has seven degrees of freedom, and we add them from the base sequentially according to the constraints of the joints. Since the panda robotic arm behaves as two claws, we set a grasping claw distance, an opening claw distance, and a grasping force at the end of the joint to facilitate subsequent grasping operations. When it is ready, the formal simulation operation can be carried out.

In the first step, in the above steps, once the relevant item distribution image is obtained in the preset camera, we call the recognition function to segment and classify the image. Perform related sorting operations on the items in the image in order from top to bottom and from left to right.

According to the above classification order, the relevant function returns a list containing garbage classification, and the classification in the list is consistent with the input order.


Fig. 4. Image of sorting operations

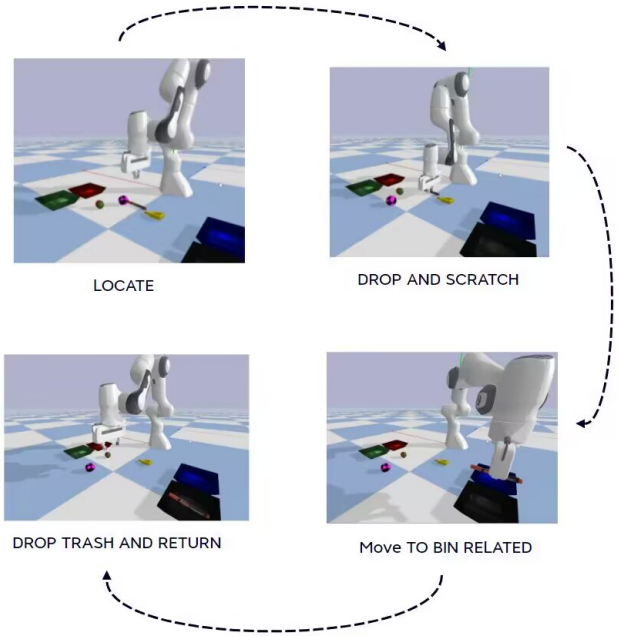


Fig. 5. The capture process

According to the classification, we use the robotic arm to grab in sequence. According to the universal classification standard, we use blue, green, red and black garbage boxes to represent kitchen waste, recyclable garbage, hazardous garbage and other garbage respectively. Similarly, the grabbing order of the robotic arm is also carried out in accordance with kitchen waste, recyclable waste, hazardous waste and other waste. Before each grab, we fix the end of the claws of the robotic arm at an initial height, and use the getBasePositionAndOrientation function to position above the origin of the corresponding garbage model according to the previously returned garbage classification order, and then we open the claws and descend to the specified Grasp objects at a high degree. Note that the center of the paw of the robotic arm will always be aligned with the center of the object on the xz plane during the descent. After the paws are closed, we raise the robotic arm to the specified height again, and translate it above the set corresponding garbage box on the xz plane, then open the paws to make the object fall, and finally return to the origin of the grab to wait for the next grab , so back and forth.

### *B. Visual Object Identification*

The input to the classification network of this project is pictures, so a composite camera is introduced in the simulation environment. In order to make the environment similar to the real application scenario, we set the task to be multiple objects, and do not consider stacking to grab. The common perception based on vision is to solve the pose of the camera coordinate system relative to the world coordinate system, that is, to describe the problem as a PnP problem (Perspective-n-point). This needs to determine the coordinates of n 3D points in the world coordinate system and the pixel coordinates of these points. In the mainstream algorithm, the minimum n is 3. However, there is no need to solve such a complex problem in the simulation environment, so we simplify this task and use the traditional image processing means to segment the image. PyBullet contains functions inside that provide the origin position of the model urdf model, and the findContours() method in OpenCV returns the contour indexing order from top to bottom and from left to right. The principle is that the method will scan the pixels of the input binary image in this order and output all points on the contours. To take advantage of this, we set up an initial scene where the items are scattered to make it easier to output the exact outline sequence. The segmented images are input into the network in order of contour to obtain classification, and the results are provided to the simulation manipulator.

The visual identification input in the simulation environment is rendered using the compositing camera built inside PyBullet. PyBullet has a built-in OpenGL GPU visualizer and a built-in CPU renderer based on TinyRenderer. This makes it very easy to render images from arbitrary camera positions. The above synthesis camera is specified by two 4*4 matrices, the view matrix and the projection matrix. The former can be specified by cv.computeViewMatrix. The parameters are three three-dimensional space coordinates, specifying the position of Eye(camera), the position of target (focus) and the up vector of the camera (used to generate the camera coordinate system). The simulation of this project uses these information to generate a composite camera. In addition, the Euler Angle of the camera can be specified to generate a view matrix. Projection matrix available cv.com puteProjectionMatrixFov generating function. The above methods use a cone perspective model to generate a projection matrix by defining Fov, aspect ratio, near plane and far plane. This method is used in this experiment. In addition, orthogonal projection can also be used. The parameters involved in the method used above are obtained by experiments in the simulation environment.

In order to adapt to the multi-object grasping task, the following operations are carried out on the obtained images in the simulation environment. After grayscale and binarization, openCV built-in function is used for contour extraction. The

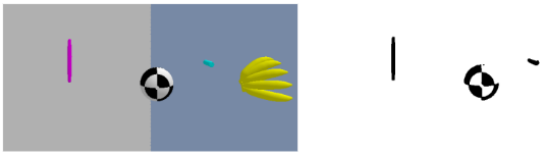

Fig. 6. Projection matrix

input of this function is binary image, and the boundary is determined by detecting the intensity mutation of pixel points. After contour filtering, the image region is intercepted according to the extreme value of the extracted contour boundary in the XY direction. Contour filtering is achieved by setting upper and lower thresholds for contour length. Since the previous contour extraction function scans from top to bottom and from left to right, the output contour index order also follows this rule. The difficulty of the above steps lies in the parameter setting of each step. Because the input and output of each step are connected, the result of parameter setting of each step will have a great impact on the final result. Parameter Settings are manually adjusted according to experimental results.

### C. Deep Learning

Waste classification has become an important national policy. Cities like Shanghai and Beijing have implemented waste classification, categorizing waste into four types: kitchen waste, recyclables, hazardous waste, and other waste. Each type is assigned a corresponding color-coded trash can. Japan has even stricter garbage classification standards, dividing waste into seven categories and specifying designated days for recycling specific types of waste. Given the prevalence of image recognition-based garbage classification methods, the rapid development of deep learning makes it an ideal approach for this task.

To develop our garbage classification model, we utilized a large dataset consisting of 56,528 RGB images from Baidu PaddlePaddle, covering 214 categories and totaling 7.13 GB. After obtaining relevant tag files from the internet, we divided the dataset into training, validation, and testing sets. Prior to training, we checked the image quality, using PIL's Image library to read and identify any errors or warnings and remove problematic images.

For data preprocessing, we performed pixel filling and resizing to 224 sizes. Data augmentation techniques such as horizontal and vertical flipping were employed using PyTorch. The processed data was then uniformly added to the dataloader.

The training phase employed the widely used ResNet50 network architecture for feature extraction. We utilized the pre-trained model of ResNet50 from Torchvision. The algorithm employed the Adam optimizer with a cross-entropy loss function, and StepLR was used for learning rate decay. Model selection was based on the highest accuracy achieved on the validation set, with a batch size of 64 and GPU memory usage of approximately 8GB.

Testing was conducted using our dataset, randomly selecting a set of photos to evaluate the network's performance. The obtained results were compared with the ground truth to calculate the network's accuracy.

After 64 epochs, the model achieved a stable accuracy of 88%. Thus, we selected the best-performing model for further tasks.

## V. Experiment

### A. Classifier Simulation Experiment

Garbage classification has now been integrated into everyone's daily life, starting from various cities in China and rapidly becoming popular nationwide. However, not everyone can follow the principles of garbage classification, so external equipment is needed for manual collaboration in garbage classification. This experiment is more similar to an example of a garbage classification model.

We use deep neural networks for simulation and aim to achieve precise recognition on different objects.

*1) Accuracy of Garbage Classification for 4 Major Categories and 216 Subcategories of Objects:* Our classification experiment achieved good results in estimating the large class level, and our classifier's test results are shown in the following figure. In the following figure, the horizontal axis represents the number of tests, which is 100 times. Each time, 100 randomly selected data from the validation set are tested, and the accuracy obtained is stable at over 95%, with the lowest test being 95

The four categories are hazardous waste, recyclable waste, kitchen waste and other waste. Although the overall accuracy rate is high, in fact, the accuracy rate of recyclable garbage and other garbage is nearly 100%, while the accuracy rate of kitchen waste is the lowest, so the network has a large bias.

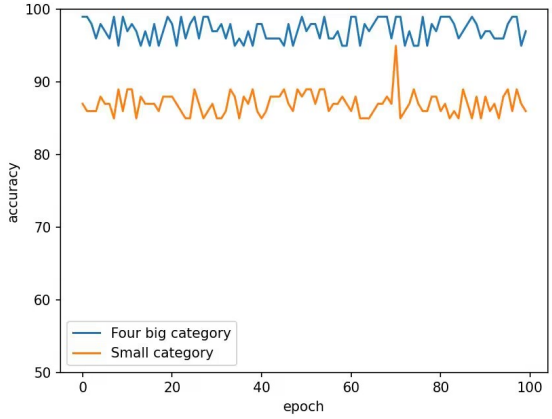


Fig. 7. Accuracy between 4 major categories and 216 subcategories

For the more than 200 subcategories subdivided under the four major categories, the accuracy has decreased significantly and remained stable at below 90% and above 85%. It indicates that objects in the same category may be classified incorrectly,

such as paper boxes and cups, in the process of large category discrimination. However, the observed advantage of large class resolution and small class resolution is the stability of the test, which does not fluctuate more than 5%. Therefore, although the network still has shortcomings in resolution, it will not exhibit significant deviations (under different conditions). It is worth mentioning that the resolution of kitchen waste is poor not only in the large category, but also in the small category. It is speculated that the number of our samples is not enough, and the order of magnitude of kitchen waste (including hazardous waste) is limited, leading to the relative weakness of these two categories. The figure shown is the comparison of accuracy between subcategories and its parent four categories.

However, the advantage of the network model is that it is relatively stable, and the upper and lower deviation is within 5%, so it is not too different because of the environmental gap. Here are the results of four of the 100 tests, with colors from left to right representing hazardous waste, kitchen waste, recyclable waste, and other waste.

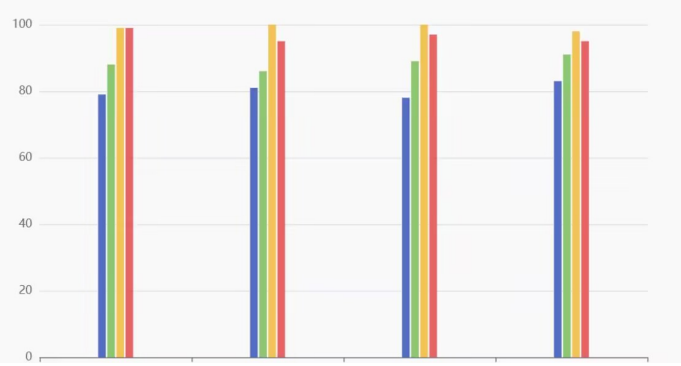


Fig. 8. Accuracy of the four garbage in different numbers of tests

### *B. Simulate the Simulation Environment Using Pybullet*

In the simulation grasping based on the panda robotic arm, we fixed the sample data at 50. It can be seen that when the shape of the grasped object is a sphere, the grasping success rate is 100%, but when the grasping object has an irregular shape, the grasping success rate is only 64%. . The analysis infers that the grasping success rate is limited by the two-claw design at the end of the claw of the panda robot arm.

Then, we analyzed 50 samples based on the distance between the captured object and the trash box, taking the square as the research object:

Taking a square object as an example, the farther the initial position of the object is from the trash box, the greater the probability of being thrown by the paw during the translation process. This is limited by the grasping structure of the two-claw robot arm and the fixed-point translation method of the robot arm.

### *C. The Visual Recognition in a RGB-photo*

The measurement criteria of visual recognition include: 1) the difference between the number of filtered sub-images and

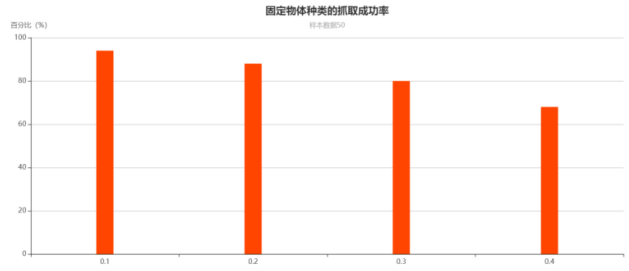


Fig. 9. Results from 50 samples

the number of actual objects. If there is a difference, the identification fails. When the same object contains two or more high-contrast colors, such as a black and white soccer ball, the same object may separate multiple Outlines, resulting in the wrong sub-image. 2) The sub-image contains the integrity of the object. If the edge color intensity of the object and the background difference is small, this part of the grayscale and the background difference is little, resulting in the sub-image contains incomplete objects. This part is difficult to pass the program inspection, so we choose fewer iterations for manual inspection. Any child image that does not fully contain an item is considered a failure.

In these cases, well-designed background is beneficial to maintain the integrity of the object contour. Some unimportant contours can be filtered out by Gaussian blurring the input image. Setting a boundary length threshold relative to the size of the input image filters out relatively small Outlines.

After optimization, the accuracy and completeness of goods segmentation have better performance. For the first criterion, we set the number of iterations as 30, 40 and 50, the size of Gaussian fuzzy convolution kernel as 3*3, 5*5 and 7*7 respectively, and the actual number of items is 4. Each test item is randomly selected from our item library. The results are as follows:

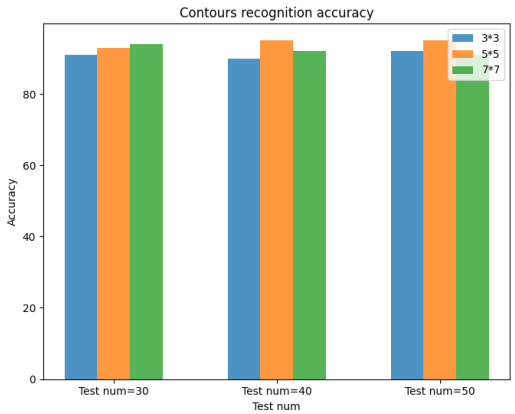


Fig. 10. Different results under different parameter

It can be seen that when the convolution kernel size is 5*5, the overall item segmentation accuracy is the highest, which effectively reduces the unimportant contour and does

not lose important boundary information. Overall, the accuracy of object segmentation is higher ( 90%).

For the second criterion, other experiments were set unchanged, and the number of iterations was reduced to 10, 15, 20 and 25. In general, since we set the background to pure white (255,255,255), the more saturated items are likely to have lower segmentation integrity. The number of sub-images isolated in this test is considered a failure if it is different from the actual number of sub-images. The results are as follows:

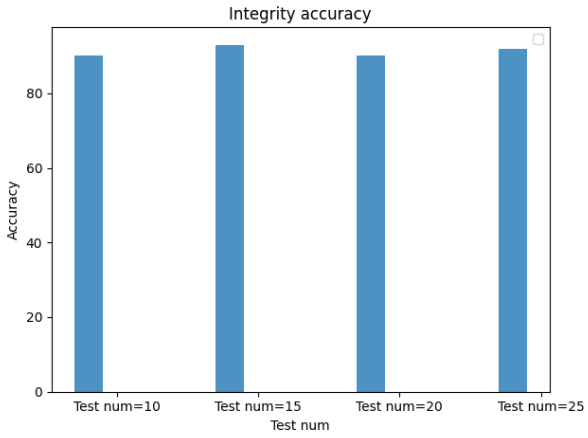


Fig. 11. The influence of changing number of iterations on accuracy

It can be seen that the accuracy of segmentation integrity is greater than 90%. If the influencing factors of the above two evaluation criteria are considered independent, it can be considered that the completeness accuracy is nearly 100%.

### *D. The Final Simulation*

We conducted a comprehensive evaluation of our project using classification, simulation, and visual segmentation. To represent different types of waste, we selected a range of objects, including a football, marker, glue, and strawberry, representing recyclable, other, hazardous, and kitchen waste, respectively. These objects were tested within the simulated environment, and the experimental results were recorded. Through a significant number of experiments, we obtained data graphs that provide valuable insights.

Among the results, errors occurred with the football due to the mechanical arm's grasping angle and posture. The marker error was caused by poor photo quality, leading to misidentification as recyclable garbage and incorrect placement in the corresponding bin. Two failures were observed with the glue. One was due to an incorrect angle in the photo, resulting in misidentification as recyclable waste. The other failure may have been caused by a blind spot in the classifier, leading to misclassification as kitchen waste. We also tested various other objects and obtained approximate accuracy rates for hazardous waste, recyclable waste, kitchen waste, and other garbage. Interestingly, the results were consistent with those obtained using the representative four objects mentioned earlier. However, it is worth noting that the accuracy was significantly affected when considering kitchen waste due to the high recognition rate of strawberries as representatives of this category. Therefore, increasing the variety of kitchen waste types resulted in a notable decrease in accuracy.

Throughout the evaluation process, we utilized our classification, simulation, and visual segmentation methods to assess the project holistically, obtaining valuable insights and understanding the strengths and limitations of our approach.

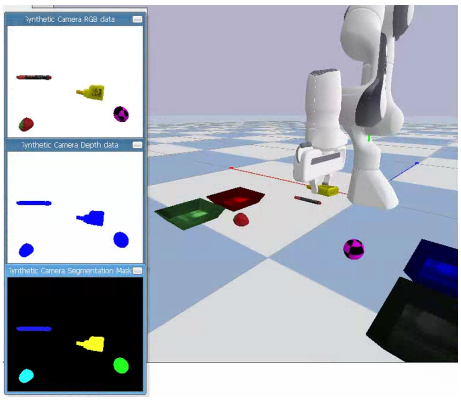


Fig. 12. The final simulation process

We then chose soccer balls and Legos, glue and mugs as spherical, rectangular, irregular and cylindrical objects to test, because in order to get the data quickly, we dropped four of the same objects at a time, and asked them to recognize and grab. The data results were as follows: we dropped each object 25 times (100 times in total). Compared with the data captured only by the robotic arm, the correctness of the data connected to the camera and the classification network will be reduced. According to the analysis, this is also related to the accuracy of the classification network itself. The accuracy of the spherical object is still nearly 100 percent, because the spherical object will not have category deviation due to the camera Angle, and it is easier to capture. The accuracy reduction of square objects is also not high, because most of the squares belong to recyclables, the accuracy is high, irregular and cylindrical objects accuracy reduction is more obvious, because the classification accuracy of harmful garbage and other garbage will be low, and the difficulty of grabbing is also high.

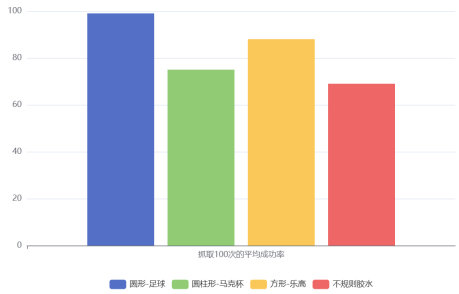


Fig. 13. The final effect of grabbing

## VI. Conclusion

The team leader of this project is Xinzi Liu , who is mainly responsible for the code and paper writing of the deep learning part. The team members include Yu taoGuo, who is responsible for the simulation module and paper writing; Sijing Qu, who is responsible for the visual segmentation module and paper writing; Haoran Wang, who is responsible for the model drawing and paper writing and integration.

During the course of this project, we successfully achieved the majority of our initial objectives, which involved integrating deep learning, computer vision, and robotic arm manipulation. We were able to accomplish the clamping of various types of garbage within the Pybullet simulation environment, resulting in a highly realistic simulation with a sorting accuracy exceeding 90%. This outcome has not only met but also surpassed our expectations for the project.

Throughout this course project, we gained valuable insights and practical experience in several key areas. Firstly, we learned how to train a neural network model using a carefully curated dataset. We explored the utilization of RGB-D cameras and OpenCV to convert object visual information into compatible network input data. Additionally, we acquired the knowledge to import Solidworks models into the simulation environment and assign appropriate physical properties to them. Moreover, we developed scripting skills to enable the manipulation of objects by the robotic arm, ensuring precise positioning within the simulated environment. The integration of these independent modules into a coherent garbage classification pipeline was a significant achievement. Through extensive experimentation, we were able to validate the accuracy and efficiency of our methodology, resulting in the successful completion of the system and the realization of several essential functions.

While we have accomplished the fundamental functionalities of garbage classification and sorting, we acknowledge the need for further refinement and enhancement. To simplify the visual identification and motion planning process, we simplified the placement mode of garbage items to a tiled configuration within the simulation environment. This approach allowed us to avoid the complexities associated with stacking objects, which are typically encountered in real-world scenarios. Additionally, we carefully selected garbage types with high recognition accuracy to mitigate sorting errors within the simulation.

Given a longer project timeline, we would endeavor to optimize the neural network, expand the training dataset, and enhance identification accuracy. Recognizing that real-world conditions often present more complex challenges than those represented within the simulation environment, we aspired to incorporate item stacking into the simulation. By leveraging depth cameras, we aimed to develop a system capable of gripping objects from shallow to deep layers, further refining the realism and practicality of our approach.

## References


[1] Ishrat Zahan Mukti and Dipayan Biswas. Transfer learning based plant diseases detection using resnet50. In *2019 4th International conference on electrical information and communication technology (EICT)*, pages 1–6. IEEE, 2019.

[2] Dhananjay Theckedath and RR Sedamkar. Detecting affect states using vgg16, resnet50 and se-resnet50 networks. *SN Computer Science*, 1:1–7, 2020.

[3] Guanhao Yang, Jintao Jin, Qujiang Lei, Yi Wang, Jiangkun Zhou, Zhe Sun, Xiuhao Li, and Weijun Wang. Garbage classification system with yolov5 based on image recognition. In *2021 IEEE 6th International Conference on Signal and Image Processing (ICSIP)*, pages 11–18. IEEE, 2021.

[4] Zhifei Zhang, Hao Wang, Hongzhang Song, Shaobo Zhang, and Jianhua Zhang. Industrial robot sorting system for municipal solid waste. In *Intelligent Robotics and Applications: 12th International Conference, ICIRA 2019, Shenyang, China, August 8–11, 2019, Proceedings, Part II 12*, pages 342–353. Springer, 2019.

[5] Chen Zhihong, Zou Hebin, Wang Yanbo, Liang Binyan, and Liao Yu. A vision-based robotic grasping system using deep learning for garbage sorting. In *2017 36th Chinese control conference (CCC)*, pages 11223–11226. IEEE, 2017.