Fruit Types Identification and Fruit Quality Classification Based on Machine Learning

Zhuoyang Ding, Xuan Tong, Tian Qiu, Xinshu Hu, Zhiqi Yin, Jingxuan Lu

***Abstract*—This project is aimed to develop a robot arm with a camera, which is able to identify different kinds of fruit, and also, the fruit quality. We suppose the robot arm can distinguish fruit with low quality from normal fruit, and change its pose to grasp the former on a moving belt to help improve agricultural automation. To make the arm recognize different objects, we use a method called vision transformers(ViTs), which has higher accuracy and efficiency than CNNs in some ways. Also, the robot arm will be driven by ROS control. This project is based on computer vision, deep learning, robot arm control, and it will achieve a robot arm grasping the target with high accuracy and efficiency.**

Key words*— *Deep learning; Fruit identification; Fruit identification; Dregs detection; Agricultural automation; Image processing

I. Introduction

In today's China, with the growth of population and the acceleration of urbanization, food safety issues in public canteens have become a key concern, such as school canteens, company canteens, large hotels and restaurants. The figures 1 are some Low-quality fruit. However, it is not an easy task to quickly and efficiently sort these ingredients. Currently, most places still hire professional chefs to complete this process. This not only costs a lot of manpower and material resources, but also requires a lot of time.

The project aims to develop a complete assembly line consisting of a conveyor belt, a sorting robot arm, a depth camera, and a high-performance core controller. The conveyor belts roll at a constant speed, transporting the purchased fruit to the bottom of the camera of the robot arm. Subsequently, when the depth camera detects the presence of fruit within its field of view, it will start taking continuous photos and transmit the captured data to the internal controller. The internal controller uses DINO-based vision transformer (VIT) algorithm that can distinguish fruit of different qualities, and send instructions to the robot arm to grasp the bad ones into nearby baskets for further processing by staff.

In short, the project aims to develop a method based on visual recognition to control the robotic arm. This method can recognize low-quality fruit in real time, allowing the robotic arm to screen out higher-quality fruit with great accuracy.

II. Problem statement

In this course project, our aim is to design a robotic arm capable of sorting fruit, discerning unwanted defective products, and removing it. We break down the task into three main components: visual recognition, machine learning, and robotic arm manipulation.

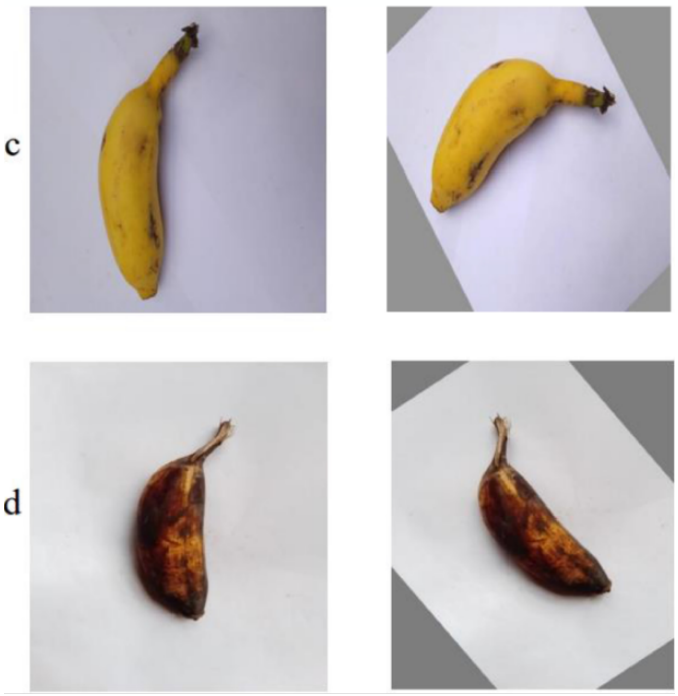


Fig. 1: The contrast between fresh and rotten fruit

To begin, we must establish communication between Python and ROS. This enables the transfer of images captured by the robotic arm to Python for initial processing and analysis. Next, we embark on training the Vision Transformer model to identify the freshness of fruit depicted in the images. For this phase, we utilize a dataset from the article An extensive dataset for successful recognition of fresh and rotten fruit [5]. Finally, we also need to build a robotic arm in the simulation environment of ROS, so that the robotic arm can sort out the defective products after receiving the judgment results.

Our goal is to achieve a robot judgment accuracy and sorting success rate of 90% or higher, with a final efficiency rating of 85% or more.

III. Literature Review

A. Machine Learning

With the development of CNNs, image-based machine learning models can be used to make the sorting and grading of agricultural products more efficient.[4] And a a simplified development procedure for image-based machine learning for visual fruit quality assessment has been presented. It is particularly suitable for domains with low availability of

both data and computational resources. [3]Figure 2gives how vision transform model is different from convolutional neural networks

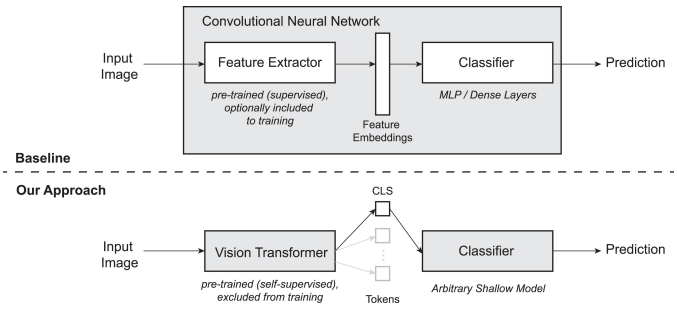


Fig. 2: Vision transformation compared to Convolutional Neural Networks

B. Fruit Grasping

For the picking of fruit, there are primarily two key considerations. The first aspect to consider is that various fruit come in different shapes and sizes, which may pose certain difficulties in gripping them. The second point is that most fruit are soft, so using conventional sharp claws may lead to damage. However, considering that most of the fruit we pick are unwanted, we prioritize the first point.

Figure 3 depicts a model of a fruit-picking robotic arm[6]. For the picking of fruit with different shapes, sensors and computer vision are used to detect and estimate the position of the fruit. Then, based on the fruit's position, the inverse kinematics of the robotic arm are calculated to position the gripper tool in front of the fruit, and the final picking method is obtained by iteratively adjusting the vertical and horizontal positions of the gripping tool through a closed-loop visual feedback[2].

Fig. 3: A robotic arm for grasping oranges

IV. Technical Approach

In this project, the main two technical problems are how to identify our target object and how to plan the trajectory of the arm.

A. Vision Transformers

We plan to use vision transformer to train the model.The vision transformer architecture comprises multiple transformer blocks, each consisting of self-attention layers and feed-forward neural networks. Initially proposed for natural language processing (NLP), transformers have been adapted for vision tasks by dividing input images into fixed-size patches, which are then linearly embedded to form token sequences. These token sequences are processed through multiple transformer layers, enabling the model to learn hierarchical representations of the input images.[1]

B. ROS Moveit

As to the motion planning, we are going to apply ROS package moveit, which provides a comprehensive set of motion planning algorithms and tools that enable robotic arms to navigate complex environments efficiently.

V. Intermediate results

A. Convolutional Neural Network

Using the train_cnn.py script[1], we have effectively trained a convolutional neural network model on a dataset comprising images of bananas, utilizing the ResNet50 architecture for the task. Throughout the training process, comprehensive performance monitoring and visualization were conducted via the Weights & Biases (wandb) platform.[2]

The loss and accuracy charts Fig.4 reveal a steady decline in loss values on both training and validation sets over time, alongside a consistent increase in accuracy, indicating the model's capability to learn from the data and improve its generalization performance. Specifically, the loss on the validation set (val_loss) decreased rapidly, and the accuracy on both training and validation (train_accuracy and val_accuracy) showed progressive improvement, signifying an effective training process and good model fit to the data.

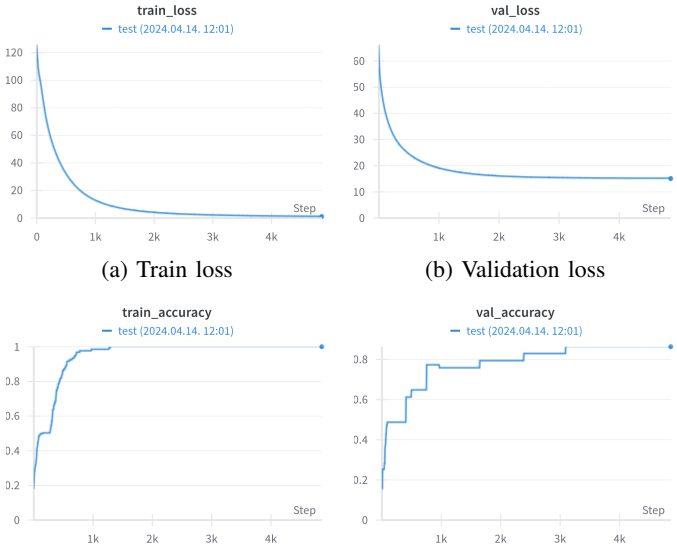
The confusion matrix Fig.5 highlighted excellent precision and recall rates for the "Green" category, representing unripe bananas, suggesting that the model performs exceptionally well in distinguishing this category. ROC Fig.6 and precision-recall curves Fig.7 further demonstrated the model's robust discriminative power across different ripeness categories.

Gradient histograms Fig.8 illustrated the distribution of gradients across different layers, with gradients remaining stable throughout training, without any signs of gradient explosion or vanishing, indicating a stable learning process.

In summary, the training of the ResNet50 architecture on the banana dataset proved to be successful, with stable training dynamics and high accuracy, demonstrating its effectiveness for image recognition tasks. This sets a solid foundation for future optimizations and applications.

[1]https://github.com/manuelknott/DINO-ViT_fruit_quality_assessment/blob/main/baseline_experiments/train_cnn.py

[2]https://api.wandb.ai/links/sustech_me336/tvwxj2w3



(a) Train loss (b) Validation loss

(c) Train accuracy (d) Validation accuracy

Fig. 4: Model training performance chart

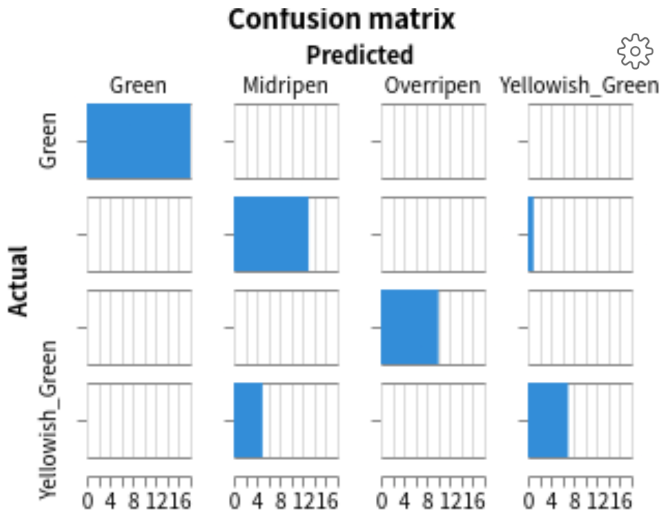


Fig. 5: Confusion matrix

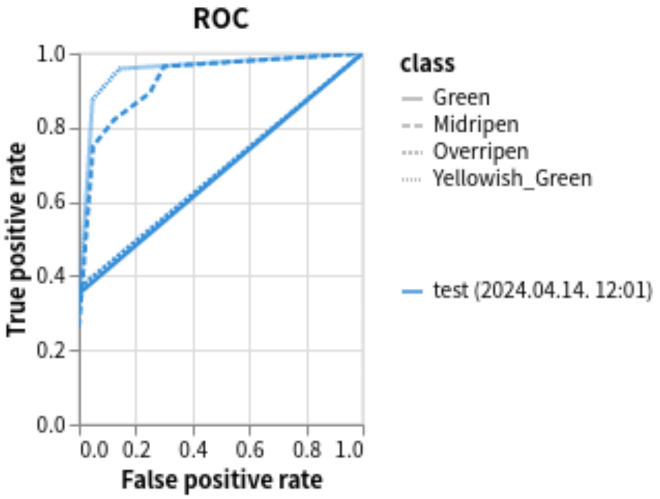


Fig. 6: Receiver operating characteristic curves

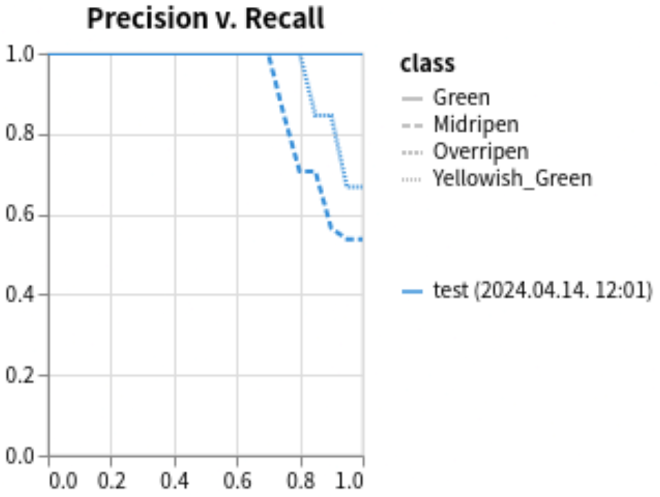


Fig. 7: Precision-Recall curves

B. Environment and Simulation

Using the codes in Github, our group has built up an environment that contains a blue basket, a convey belt and has red cubes moving on it every several seconds. Our group uses a ur5 manipulator with a depth camera and the manipulator can identify the red cubes and catch them, and throw them into the basket, which is similar to our purpose. Figure 9shows the working manipulator.

After the robotic arm's built-in camera is activated, it can take photos according to our requirements. Simultaneously, Python image processing code will also be initiated, waiting for the image to be received. Currently, the set interval for taking photos is 0.5 seconds. The current code is a simulated demo used to verify if the communication effect can be achieved. Whenever the camera takes a photo, Python will read this image and process it into black and white, placing it into another file package. Figure 10is the implementation effect of the demo, subsequent code will be added to integrate with the ViT code.

VI. Future Plans

The future plans of our project are as follows:

1) Further training the model with ViT, comparing the ViT result with CNN.
2) Changing the red cubes, trying to insert some pictures on top of red cubes.
3) Using the machine learning model to train the manipulator, making it distinguish the defective from normal fruit.

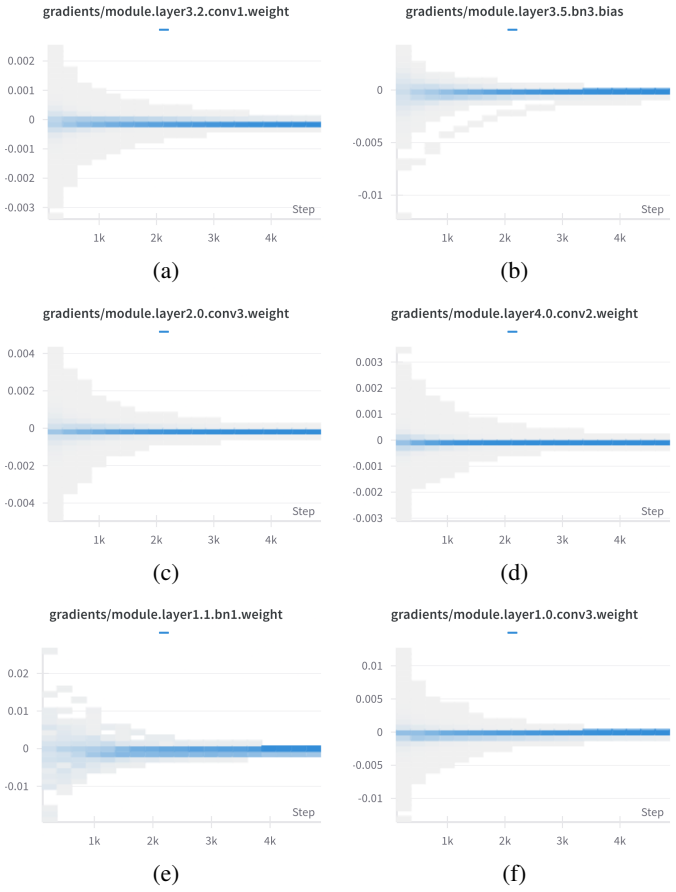


Fig. 8: Training gradients distribution visualizations

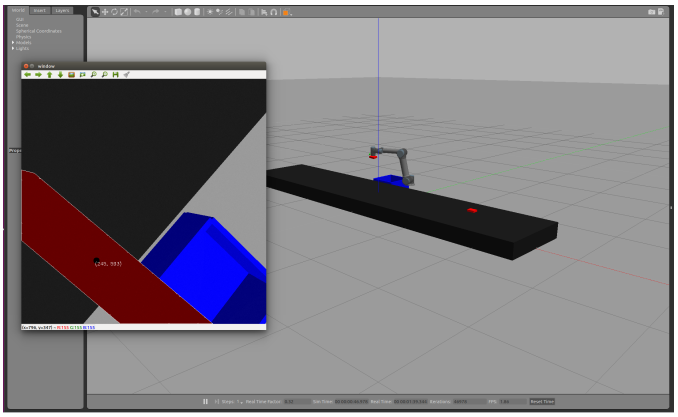


Fig. 9: Manipulator grasping red cube

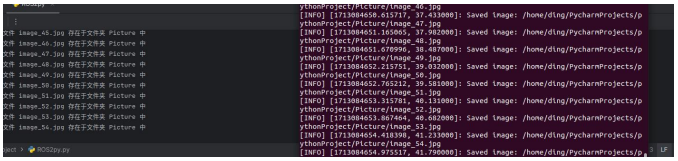


Fig. 10: Communication between ROS and Python

ACKNOWLEDGMENTS

REFERENCES

[1] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[2] Davinia Font, Tomàs Pallejà, Marcel Tresanchez, David Runcan, Javier Moreno, Dani Martínez, Mercè Teixidó, and Jordi Palacín. A proposal for automatic fruit harvesting by combining a low cost stereovision camera and a robotic arm. *Sensors*, 14(7):11557–11579, 2014. ISSN 1424-8220. doi: 10.3390/s140711557. URL https://www.mdpi.com/1424-8220/14/7/11557.

[3] Manuel Knott, Fernando Perez-Cruz, and Thijs Defraeye. Facilitated machine learning for image-based fruit quality assessment. *Journal of Food Engineering*, 345:111401, 2023. ISSN 0260-8774. doi: https://doi.org/10.1016/j.jfoodeng.2022.111401. URL https://www.sciencedirect.com/science/article/pii/S0260877422004551.

[4] Ricardo Ribani and Mauricio Marengoni. A survey of transfer learning for convolutional neural networks. In *2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)*, pages 47–57, 2019. doi: 10.1109/SIBGRAPI-T.2019.00010.

[5] Nusrat Sultana, Musfika Jahan, and Mohammad Shorif Uddin. An extensive dataset for successful recognition of fresh and rotten fruits. *Data in Brief*, 44:108552, 2022. ISSN 2352-3409. doi: https://doi.org/10.1016/j.dib.2022.108552. URL https://www.sciencedirect.com/science/article/pii/S2352340922007594.

[6] Takeshi Yoshida, Yuki Onishi, Takuya Kawahara, and Takanori Fukao. Automated harvesting by a dual-arm fruit harvesting robot. *ROBOMECH journal*, 9(1), 9 2022. doi: 10.1186/s40648-022-00233-9. URL https://doi.org/10.1186/s40648-022-00233-9.