Project Report for Kiwifruit Harvest Project (System and Methodology)

Kiwi Group

January 28, 2024

TLDR

Too Long Don't Read (TLDR): In this pdf., we report 1) the background, 2) motivation, 3) robotic system and 4) algorithm of our project. **The experiment procedure and replication notes are in other files.**

1 Introduction

Conventional fruit harvesting relies heavily on human labor, exposing workers to pain, monotonous tasks, and potential dangers. For example, kiwifruit cultivation is widespread across various countries, with a significant reliance on seasonal manual labor for harvesting. This underscores the economic justification for developing a kiwi harvesting robot, aiming to alleviate the challenges associated with human-intensive harvesting practices. Therefore, field harvesting robots are in demand. Nonetheless, substantial challenges persist over computer, mobility, control and manipulation technologies [BBRvH16, ZWA+22] in this area. **Therefore, fruit harvesting is a practical, motivating and challenging mobile manipulation topic.**

However, fruit-harvesting robot as a course project represents a multifaceted challenge The complexity of this project arises from many key aspects, including:

- **Environment Uncertainty:** Unlike industrial robots operating, fruit-harvesting robot operates in unstructured environments. It demands robust and adaptable algorithms.
- **Vision-Based Decision Making:** The robot should move and pick fruits by synchronizing vision and motion control. path planning, obstacle avoidance, and decision-making under uncertainty.
- **Software and Hardware Integration:** The project demands a high level of integration between various software components (like image processing, control algorithms) and hardware (sensors, grippers, robotic arms).

In conclusion, developing a fruit-picking robot encapsulates a comprehensive range of engineering challenges. This project not only tests 1) our technical skills but also 2) our ability to integrate diverse systems into a cohesive and functioning whole.

1.1 Our Approach

We use Bunker platform and Dobot CR5 arm to build a kiwi harvesting robot. We use an NUC with ubuntu 20.04 as the computation host. The image signals are captured by realsense RGBD camera. We process the collected point cloud data based on Grasp Pose Detection (GPD) module. We use RRT to plan the arm, supported by MoveIt!. We use an additional script to control the gripper.

In CV module we base on the GPD module. It is a lightweight grasp detection pipeline that can run in an NUC. The pipeline includes a CNN network. It generates thousands of grasps, then select less than 100 grasp candidate, use the CNN to score and return the grasp pose that has highest score. We communicate the information with ROS. In the project, scripts from rospy and C++ is integrated.

1.2 Related Works

Recent work [WJN+19] uses a "tube" to carry the picked kiwi instead of pick-and-place. Another work [LFY+15] examines the performance of different end effectors. In comparison, we use gripper to harvest fruits, which needs more precious motion planning and delicate force control.

Current methods use heavy reconstruction algorithms to optimize the performances. For instance, [SZW+21] segments the canopy and reconstruct the wire. However, the works could need too much computation resources, and we build a lightweight pipeline. Another current approach is to integrate image segmentation technology ([WJN+19]), but our approach is end-to-end, can produce grasps without such an intermediate stage.

2 System Description

Our robotic system is a mobile robot shown in Fig. 1. It is an assembly of four key components:

- **NUC:** This serves as the computational host of our system. It's a compact computer that provides the necessary processing capabilities for complex tasks and algorithms.
- **Bunker:** This is our mobile platform. It is holonomic. Its design allows for navigation and stability during operations.
- **Robot Arm:** We have a six DoFs arm. It enables multi-directional movement. The arm can position and orient the gripper in a wide range of configurations, crucial for tasks requiring high accuracy.
- **Two-Finger Gripper:** We have a two-finger parallel gripper at the end of robot arm, we can control the speed, force, and position (in one direction) of the two fingers.
- **LIDAR and Depth Camera:** We have a LIDAR to map and navigate, and a RGBD camera for real-time point cloud retrival.

Figure 1: Our robotic System.

3 Method

Our method has four main components 1) point cloud retrival and analysis, 2) grasp pose generation, 3) coordinate transformations, 4) motion planning. In the next two subsections, we introduce 1), 2) and 3), 4) respectively.

3.1 Computer Vision and Grasp Detection

In our kiwifruit harvesting project, determining grasp pose is crucial. It enables the robotic system to adapt to varying sizes and orientations of kiwifruits, which is vital for maintaining the quality and integrity of the fruit during harvesting. We utilize Convolutional Neural Network (CNN) ([LB+95]) to process 3D point cloud data, and generate reliable grasp proposals in the framework of Grasp Pose Detection (GPD) [TPGSP17]. We show the visualize of candidate grasp poses in Fig. 2. The grasp candidates are concentrated around the kiwifruit to grasp, and the deeper color represents higher score.

Our method composes of five procedures:

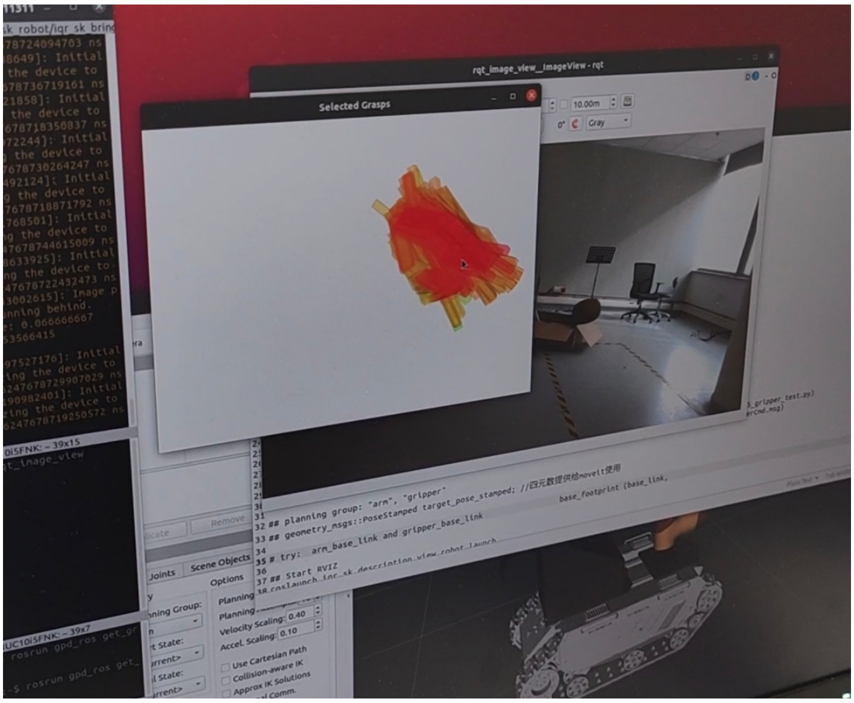

Figure 2: Grasp Proposal Visualization.

- **Preprocessing:** The (colored) point cloud data, obtained from Realsensedepth-sensing camera, is preprocessed to remove noise and outliers. This step include 1) voxelization to reduce the data size and 2) filtering to focus on regions of interest (ROIs).

- **Candidate Generation:** Then the algorithm generates a large set of grasp candidates (typically 2000 to 4000 ones) by sampling numerous points in the point cloud. Each candidate represents a hypothetical grasp pose. This sampling is exhaustive to ensure a comprehensive exploration of possible grasps. The candidates are represented defined by its position and orientation in the 3D space.

- **Feature Extraction:** For each grasp candidate, a set of descriptive features is extracted. These features encapsulate local and global information of the grasp: geometric properties, gripper orientation, and the local curvature of the object surface.

- **Classification:** We use a pretrained classifier. It evaluates each candidate's viability based on the extracted features. It assigns a score reflecting the likelihood of successful grasping, ranking the candidates, predicting the likelihood of each grasp's success.

- **Post-processing:** The top-ranked grasp poses (less than 100 ones) are filtered and refined to adapt to the robotic manipulator. We check for collisions and ensuring kinematic reachability. Finally, we get the gripper coordinate for a final high-scored and feasible grasp pose.

3.2 Grasp Motion Planning and Gripper Control

3.2.1 Coordinate Transformation

To facilitate the transformation of the gripper coordinates into the camera coordinate system, we use a rospy script. This script converts the gripper's coordinates into the grasp center point coordinates within the camera's frame of reference. These coordinates comprise a three-dimensional position vector and a rotation quaternion. The usage of ROS's `tf` library enables us to perform coordinate transformations between different frames effectively. This capability is essential for maintaining accuracy and consistency in our system's spatial awareness.

3.2.2 Integration with MoveIt! and Motion Planning

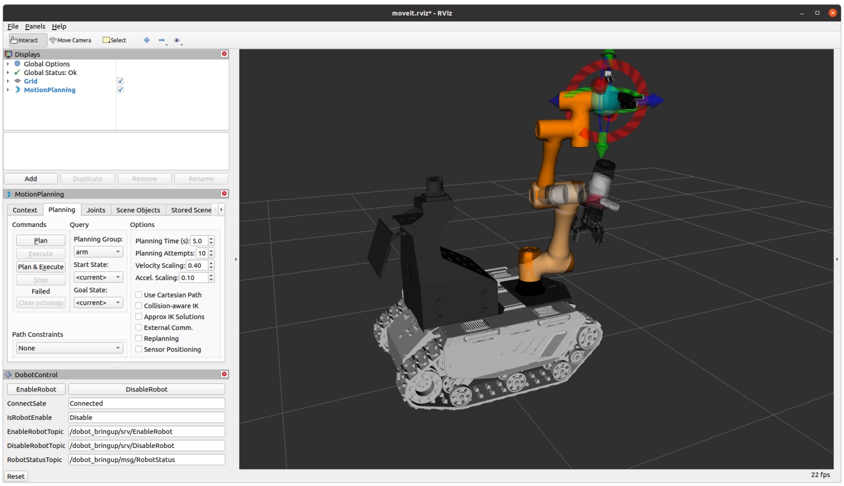

Figure 3: MoveIt! and rViz

Our system utilizes MoveIt! to control the position of the Dobot CR5's end-effector. Its GUI is Fig. 3. MoveIt plan the robot arm based on the "base footprint" coordinate system. We transform the coordinates from the camera coordinate system to the "base footprint" coordinate system. Once transformed, these coordinates are relayed to the motion planner. For motion control, our system employs the Rapidly-exploring Random Tree (RRT) algorithm. This algorithm is known for its efficiency in planning and executing complex motion paths, especially in environments with obstacles or specific constraints. Then we control the gripper with one seperate rospy script. We use 0.1 N force to grasp a fruit.

Notes

The replication procedures are in rep.txt, and also briefly sketched in index.md. The scripts for replications are in tips.txt. *Our robot arm could fail in communications with the Next Unit of Computing (NUC). If it errs and can not move, we should restart until the "Dobot Control" panel of RVIZ can have "Enabled" and the light of robot arm becomes green at the same time. More details are in rep.txt.*

References

[BBRvH16] Marcel Bergerman, John Billingsley, John Reid, and Eldert van Henten. Robotics in agriculture and forestry. *Springer handbook of robotics*, pages 1463–1492, 2016.

[LB+95] Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[LFY+15] Fu Longsheng, Zhang Fanian, Gejima Yoshinori, Li Zhen, Wang Bin, and Cui Yongjie. Development and experiment of end-effector for kiwifruit harvesting robot. *Nongye Jixie Xuebao/transactions of the Chinese Society of Agricultural Machinery*, 46(3), 2015.

[SZW+21] Zhenzhen Song, Zhongxian Zhou, Wenqi Wang, Fangfang Gao, Longsheng Fu, Rui Li, and Yongjie Cui. Canopy segmentation and wire reconstruction for kiwifruit robotic harvesting. *Computers and Electronics in Agriculture*, 181:105933, 2021.

[TPGSP17] Andreas Ten Pas, Marcus Gualtieri, Kate Saenko, and Robert Platt. Grasp pose detection in point clouds. *The International Journal of Robotics Research*, 36(13-14):1455–1473, 2017.

[WJN+19] Henry AM Williams, Mark H Jones, Mahla Nejati, Matthew J Seabright, Jamie Bell, Nicky D Penhall, Josh J Barnett, Mike D Duke, Alistair J Scarfe, Ho Seok Ahn, et al. Robotic kiwifruit harvesting using machine vision, convolutional neural networks, and robotic arms. *biosystems engineering*, 181:140–156, 2019.

[ZWA+22] Hongyu Zhou, Xing Wang, Wesley Au, Hanwen Kang, and Chao Chen. Intelligent robots for fruit harvesting: Recent developments and future challenges. *Precision Agriculture*, 23(5):1856–1907, 2022.