Bin picking

Nan Jiang

Stuttgart University st181029@stud.uni-stuttgart.de

Abstract. In this project, a robotic arm and its corresponding task environment were constructed within the Gazebo simulation environment. The robotic arm was integrated with the Moveit! framework for motion planning and control. Additionally, the YOLOv5 [4] vision framework was incorporated to enhance the robotic system's perception capabilities. The combined system successfully completed a bin picking [1] task, demonstrating the effectiveness of the integration. This work highlights the potential of using advanced simulation environments and state-of-the-art vision frameworks to develop and test robotic applications in a controlled and efficient manner.

Keywords: Gazebo · Moveit! · YOLOv5.

1 Motivation

The motivation for this project is to enhance automation, quality inspection, efficiency, and safety in industrial environments. Modern manufacturing demands automation for competitiveness. The robotic arm, integrated with the Moveit! framework, automates repetitive tasks, increasing throughput and reducing human error, leading to consistent production rates. Ensuring product quality is vital; the YOLOv5 vision framework and RGBD camera enable real-time inspections, detecting defects and verifying components to ensure high-quality products. Efficiency is boosted by the robotic arm's continuous operation, with the vacuum gripper facilitating quick handling and optimized movements reducing cycle times. Safety is improved by using robots for hazardous tasks, reducing the risk of accidents. The robotic arm operates safely alongside humans with sensors to avoid collisions, minimizing exposure to dangerous conditions. This project leverages robotics and vision technologies to automate processes, improve quality, boost efficiency, and enhance safety, showing significant potential for innovation in manufacturing.

2 Environment

2.1 Robotic Arm

The implementation of the robotic arm in this project involved constructing a seven-degree-of-freedom (7-DOF) robotic manipulator. Each of the seven rotational joints was meticulously designed to provide precise and flexible movement, ensuring the arm could achieve a wide range of positions and orientations required for complex tasks.

The end effector of the robotic arm was equipped with a vacuum gripper. This choice was driven by the need for a versatile and reliable gripping mechanism capable of handling various objects with different shapes and sizes. The vacuum gripper was chosen for its ability to securely pick up items through suction, making it particularly suitable for bin picking tasks where the objects to be handled can vary significantly.

The combination of a seven-degree-of-freedom design and a vacuum gripper end effector equipped the robotic arm with the flexibility and capability needed to perform a wide range of manipulation tasks, making it a robust solution for industrial automation applications.

2.2 RGBD Camera

The RGBD camera system was constructed using a Gazebo plugin to create a comprehensive vision system. This camera is crucial for integrating with the YOLOv5 framework, providing the necessary image inputs for object detection and recognition tasks.

The RGBD camera captures both color (RGB) and depth (D) information, which significantly enhances the perception capabilities of the robotic system. The depth information allows the system to accurately gauge the distance and dimensions of objects, facilitating more precise and efficient object handling.

In Gazebo, the RGBD camera was configured to simulate real-world camera parameters and environmental conditions, ensuring that the visual data acquired is realistic and useful for subsequent processing. The integration with YOLOv5 enables real-time object detection and classification, allowing the robotic arm to identify and locate objects within its workspace dynamically.

3 Controlling and Planning

3.1 Controller Setup

PID (Proportional-Integral-Derivative) controller is utilized for precise control of the robotic arm's movements. The PID controller continuously calculates an error value as the difference between a desired setpoint and a measured process variable, and applies corrections based on proportional, integral, and derivative terms. The PID controller parameters (proportional gain, integral gain, and derivative gain) were carefully tuned [5] through iterative testing to achieve a balance between responsiveness, stability, and precision. Implemented within the Gazebo simulation environment, the PID controller enables real-time adjustments and testing under various conditions, ensuring accurate trajectory following and stability.

This controller setup was accomplished using the Gazebo plugin, which provides an interface for defining and controlling the PID parameters for each joint of the robotic arm. The Gazebo plugin allows for seamless integration and simulation of the PID controller within the virtual environment, facilitating the fine-tuning of control parameters and ensuring that the robotic arm can operate effectively under simulated conditions before being deployed in real-world applications.

Table 1. PID Gains for Robotic Arm Joints

Joint	Proportional Gain (P)	Integral Gain (I)	Derivative Gain (D)	Integral Clamp (I_clamp)
Joint 1	10000.0	100	0.1	0.0
Joint 2	10000.0	100	0.1	0.0
Joint 3	10000.0	100	0.1	0.0
Joint 4	10000.0	100	0.1	0.0
Joint 5	10000.0	100	0.1	0.0
Joint 6	10000.0	100	0.1	0.0
Vacuum Gripper Joint	10000.0	100	0.1	0.0

3.2 Planning

The planning component of the robotic arm's control system is primarily handled by the Moveit! framework, which offers a robust set of motion planning algorithms [2] [3]. Moveit! is a widely used software for robotic motion planning, providing advanced capabilities for path planning, manipulation, kinematics, and collision detection.

The integration of Moveit! with the robotic arm allows for sophisticated motion planning and execution, ensuring that the arm can navigate complex environments, avoid obstacles, and perform tasks with high precision. Moveit! also provides tools for visualizing and debugging the planned paths, enabling real-time adjustments and optimizations.

Planning Scene The planning scene is visualized using RViz, a powerful 3D visualization tool that is part of the Robot Operating System (ROS) ecosystem. RViz provides an intuitive interface for visualizing the planning environment, the robotic arm's movements, and other essential elements of the system. RViz allows users to create and visualize the entire planning scene, including the robot model, obstacles, and goal positions, helping to understand spatial relationships and planning constraints in real-time. Additionally, RViz can display the planned and executed trajectories of the robotic arm, enabling users to verify the correctness and efficiency of motion plans, which aids in debugging and optimizing the motion planning process. The tool can also display image data from the RGBD camera, providing a real-time view of the robot's perception, crucial for vision-based control and object recognition tasks. Furthermore, RViz interfaces with various ROS topics and services, allowing for dynamic interaction with the robot and the environment. Users can monitor and control different aspects of the robotic system through these topics and services, facilitating seamless integration and control. By using RViz to visualize the planning scene, trajectories, and sensor data, developers gain a comprehensive overview of the robotic system's operation, enhancing the development and testing process with real-time feedback and insights, and ensuring that the robotic arm performs accurately and efficiently in its tasks.

4 Vision System

The vision system is an integral part of the robotic arm's functionality, enabling it to perform tasks such as object detection and recognition. By integrating the camera's published topics into a trained YOLOv5 model, the system can achieve real-time target detection, enhancing its ability to interact with the environment dynamically. The vision system setup involves several key components, including data acquisition, model training, camera calibration, and detection.

4.1 Data Acquisition

The first step in developing an effective vision system is acquiring a diverse and representative dataset. For this project, we continuously adjusted the camera's pose to capture images of objects from different angles and distances. The RGBD camera, suspended within the Gazebo environment, collected a total of 200 images, ensuring that the dataset included a variety of perspectives. The RGBD camera captures both color (RGB) and depth (D) information, providing a comprehensive dataset that enhances the training process. This approach ensures robustness by capturing objects under different conditions, making the model more adaptable to various scenarios. Additionally, we manually annotated the dataset, labeling each image to identify and classify the objects within it. This labeling process is crucial for training the YOLOv5 model, as it provides the ground truth needed for the model to learn to detect and recognize objects accurately.

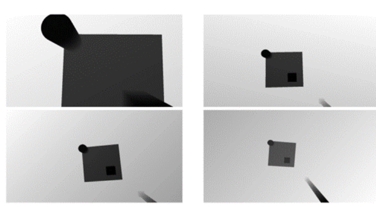

Fig. 1. Data Acquisition: RGBD camera capturing images from various angles and distances in the Gazebo environment.

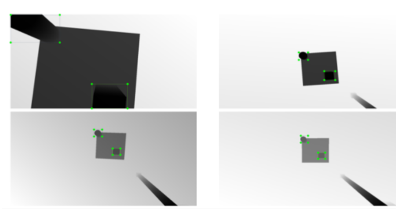

Fig. 2. Data Annotation: Manually labeling the dataset to identify and classify objects.

4.2 Model Training

Once the dataset is collected, it is used to train the YOLOv5 model. YOLOv5 (You Only Look Once, version 5) is a state-of-the-art object detection model known for its speed and accuracy. We started with a pre-trained YOLOv5 model, which was fine-tuned using our dataset. This pre-trained model already has a solid understanding of basic object features, which significantly reduces the training time and improves the model's accuracy. The training process involves feeding the dataset into the model, which learns to identify and locate objects within images. The training is done using a high-performance computing environment to ensure efficiency and effectiveness.

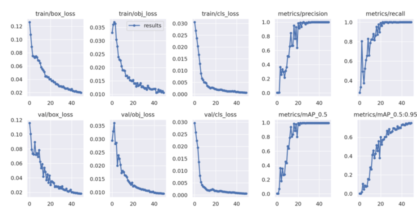

Fig. 3. Model Training: Fine-tuning the pre-trained YOLOv5 model with the annotated dataset.

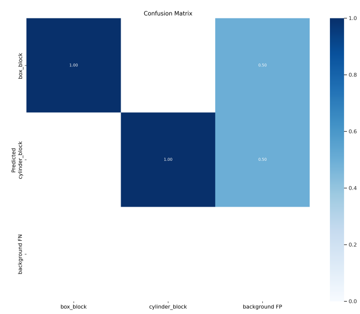

Fig. 4. Confusion Matrix: Evaluating the performance of the trained YOLOv5 model.

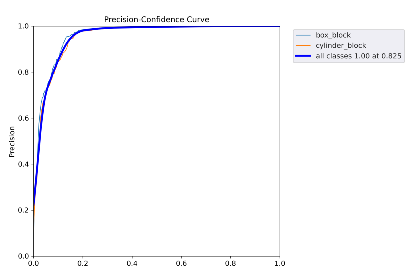

Fig. 5. Precision-Confidence Curve: Analysis of the precision at various confidence levels.

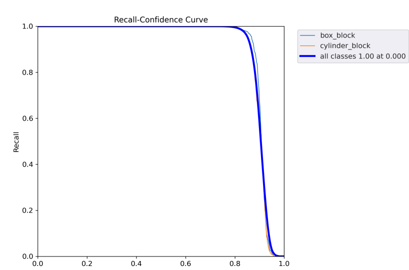

Fig. 6. Recall-Confidence Curve: Analysis of the recall at various confidence levels.

4.3 Camera Calibration

Camera calibration is essential to ensure accurate depth and spatial measurements. The calibration process involves determining the camera's intrinsic parameters, such as focal length and optical center, as well as its extrinsic parameters, such as orientation and position within the Gazebo environment. Calibration is performed using a set of standard techniques and tools provided by the ROS ecosystem. Accurate calibration ensures that the visual data aligns correctly with the robot's coordinate system, enabling precise object detection and manipulation.

4.4 Detection

With the camera calibrated and the YOLOv5 model trained, the vision system can perform real-time object detection. The camera continuously publishes image data to a ROS topic, which is then processed by the YOLOv5 model to detect and locate objects within the scene. Detected objects are annotated with bounding boxes and class labels, providing the robotic arm with the necessary information to interact with the objects. This real-time detection capability is crucial for tasks such as bin picking, where the robot must identify and grasp objects quickly and accurately.

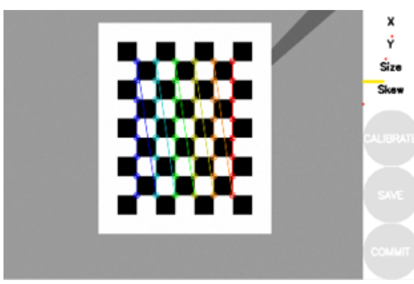

Fig. 7. Camera Calibration: Setting intrinsic and extrinsic parameters for accurate depth and spatial measurements.

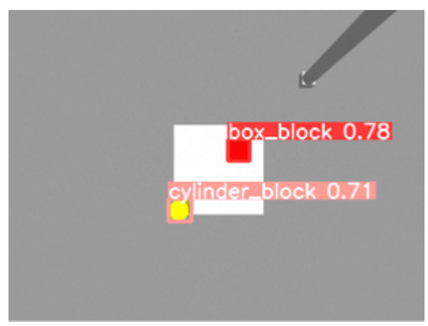

Fig. 8. Real-time Detection: The YOLOv5 model detecting and annotating objects with bounding boxes and class labels.

5 Integrated System and Bin Picking Task

The final stage of this project involved integrating the vision system, control system, and planning system within the Gazebo environment to accomplish the bin picking task. This integration allows the robotic arm to perform complex manipulation tasks autonomously by combining real-time object detection with precise motion control and planning. In the Gazebo simulation environment, we integrated the RGBD camera's data with the YOLOv5 detection model, the PID controller for movement precision, and the Moveit! planning framework for path planning. The primary task for the integrated system was to perform bin picking, specifically targeting red blocks and placing them into a blue bin. The process involved several key steps: the RGBD camera continuously captures the scene and publishes image data to a ROS topic. The YOLOv5 model processes this data in real-time to detect objects, particularly identifying red blocks among other items. Once a red block is detected, the system calculates its position relative to the robotic arm using the depth information provided by the RGBD camera. This position data is crucial for accurate object manipulation. Using the Moveit! framework, a collision-free path is planned for the robotic arm to move towards the detected red block. The planning algorithms ensure that the arm can reach the block efficiently without colliding with other objects or the environment. The PID controller guides the robotic arm to the calculated position above the red block. The vacuum gripper then activates, securely attaching to the block. After successfully grasping the red block, the system plans a new path to move the arm towards the designated blue bin. The arm then places the red block into the blue bin, completing the bin picking task.

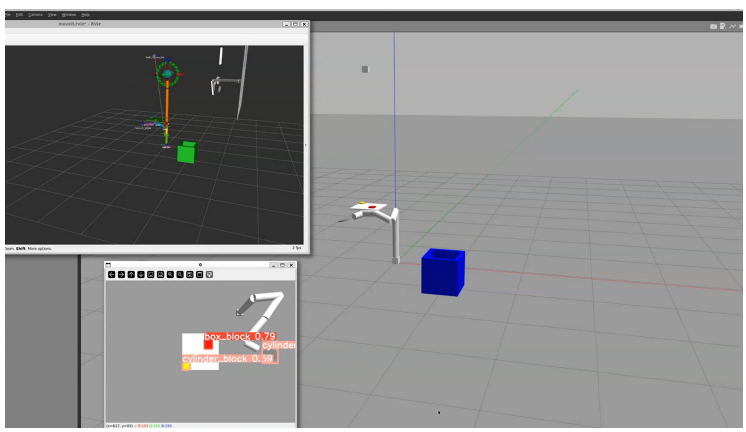

Fig. 9. Integrated System: The robotic arm performing the bin picking task by moving a red block to a blue bin within the Gazebo environment.

6 Limitations and Future Work

6.1 Limitations

While the integrated system demonstrates the potential for robotic automation in bin picking tasks, there are several limitations to the current setup. One significant limitation is that the vacuum gripper can only handle small objects. The size of the suction cup restricts the range of objects that can be effectively picked and placed, limiting the system's applicability to a narrow range of tasks. Additionally, the task environment used in this project is relatively simple, involving only two object categories (red block and yellow cylinder). This simplicity does not fully represent the complexity of real-world industrial scenarios, which often involve a wide variety of objects and more intricate manipulation tasks.

6.2 Future Work

Future work could address these limitations by expanding the system's capabilities and exploring more complex tasks. One potential improvement is the implementation of a gripper mechanism to handle a broader range of objects, including those that are too large or irregularly shaped for the vacuum gripper. This would significantly enhance the system's versatility and applicability in diverse industrial settings. Another direction for future work is to consider more complex task environments that involve multiple object categories and require real-time multi-class object recognition. This would better simulate real-world conditions and push the boundaries of the current system's capabilities, making it more robust and adaptable to various industrial applications.

In summary, while the current system successfully demonstrates an integrated approach to robotic bin picking, future enhancements could focus on extending its object handling capabilities and tackling more complex and varied task environments. These advancements would further solidify the system's potential for widespread use in industrial automation.

References

1. Jiaxin Guo, Lian Fu, Mingkai Jia, Kaijun Wang, and Shan Liu. Fast and robust bin-picking system for densely piled industrial objects. In *2020 Chinese Automation Congress (CAC)*, pages 2845–2850. IEEE, 2020.
2. Sertac Karaman, Matthew R Walter, Alejandro Perez, Emilio Frazzoli, and Seth Teller. Anytime motion planning using the rrt. In *2011 IEEE international conference on robotics and automation*, pages 1478–1483. IEEE, 2011.
3. Sunil Kumar and Afzal Sikander. A modified probabilistic roadmap algorithm for efficient mobile robot path planning. *Engineering Optimization*, 55(9):1616–1634, 2023.
4. Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
5. Muhammad Hamza Zafar, Hassaan Bin Younus, Syed Kumayl Raza Moosavi, Majad Mansoor, and Filippo Sanfilippo. Online pid tuning of a 3-dof robotic arm using a metaheuristic optimisation algorithm: A comparative analysis. In *International Conference on Information and Software Technologies*, pages 25–37. Springer, 2023.