

Zhenxiong Tang - Individual Progress Report

****Course:**** ECE 445 Senior Design Laboratory

****Project:**** A Vision-Integrated Robot for Autonomous Book Classification in Library Environments

****Team:**** Team #34

I. Introduction

My primary responsibility in this project has been the control system, decision logic, and main control program. Our team's system combines perception, planning, and robotic manipulation to automate the process of returning books from a collection bin to the correct shelf. In this architecture, the perception subsystem identifies books and detects shelf gaps, the motion subsystem physically moves the robotic arm and gripper, and my control subsystem sits between them. My job is to receive perception data, maintain system state, compute target coordinates, and send commands to the motion subsystem in the correct order.

This role is important because our system cannot rely on fixed coordinates. The camera is mounted on the arm, so detections are relative to the current camera pose rather than a permanent world frame. As a result, the controller must track the current gripper position, infer the camera position, convert relative detections into world coordinates, and ensure that the arm does not move to the next step until the previous movement has completed. Up to this point, I have completed the software architecture for this control pipeline and validated it in a mock environment.

II. Design Work

The main improvement I made was redesigning the earlier linear prototype into a state-machine-based controller. The controller now moves through explicit stages such as global bin scan, task selection, target localization, picking, shelf scanning, placement, tilt checking, and return to the home pose.

I also separated the program into several layers. The task model stores what books still need to be returned, while the world model stores dynamic information gathered during runtime, such as detected books, detected shelf layers, and updated shelf occupancy estimates. This separation is useful because task intent stays relatively stable, while perception results may change from scan to scan.

Another major part of my work was implementing coordinate transformation. Since the camera and gripper centers are different, the controller must use hyperparameters such as camera offset, gripper orientation, scan arc, and arm length to compute the current camera pose and convert camera-relative perception outputs into world-referenced motion targets. I also implemented scan-path generation for both the return bin and the shelf area.

For decision logic, I implemented book-picking and shelf-placement planning. During picking, the controller uses the detected left edge, right edge, and depth of a book to estimate a pick pose while accounting for gripper insertion depth. During placement, it searches for a shelf gap large enough for the target book, computes a safe approach pose, and then computes a final release pose.

To support team integration, I created dedicated adapter layers for perception and motion. These adapters define exactly what data fields the other subsystems must provide and include Chinese comments so my teammates can connect their modules without needing to understand the entire controller implementation.

III. Verification

So far, I have completed software-level verification using mock data, but not physical hardware testing. In the mock environment, the controller can execute the full workflow after the user enters the required hyperparameters. The tested sequence includes scanning the return bin, creating tasks from detected titles, re-localizing a target book, computing a pick pose, sending movement and gripper commands, scanning shelves, selecting a gap, computing placement poses, placing the book, and moving on to the next task.

In the current mock setup, I tested the controller with three books, and the system successfully completed the entire sequence for all of them. This verified several key points: the program can parse and use hyperparameters correctly, the control logic can maintain pose-stamped observations, the task queue can advance properly, the coordinate-conversion pipeline is internally consistent, and the controller waits for motion acknowledgments before continuing. I also verified that the tilt-check branch is reachable and that the controller can request operator input when a shelf observation indicates a tilted book.

However, several important tests have not yet been performed. The controller has not yet been connected to the real perception pipeline, the real motion subsystem, or the physical robotic arm. Because of this, I do not yet have quantitative hardware results such as grasp success rate, placement accuracy, or real timing performance. I also still need to expand exception handling for cases such as failed detections, missing gaps, or failed motion acknowledgments.

IV. Conclusion

Overall, I believe my progress is on schedule. I have completed the main software structure needed for the control, planning, and coordination side of the project. The system is not yet hardware-complete, but it is software-complete enough to support integration with the remaining team modules.

My plan for the rest of this week is to continue refining those interfaces and improving

robustness. The other teammates responsible for the perception and motion subsystems are expected to finish their modules during the same period. Next week, the major goal is to connect the controller to the robotic arm and begin real hardware testing.

From an ethical perspective, this project requires honesty about system capability and caution in controlling physical hardware. Since the robot will eventually operate in a real environment, it is important that the controller not issue unsafe commands or claim successful operation without proper acknowledgment. For that reason, I have tried to design the controller around explicit confirmation of completed actions and to clearly distinguish between mock verification and real hardware validation. I believe this is both good engineering practice and an ethical responsibility.