

# Individual Progress Report - Xinrui Xiong

ME470 Senior Design

Team 34: A Vision-Integrated Robot for Autonomous Book Classification

## Introduction

### Team Project Overview

Team 34 is developing an autonomous book reshelving system for library environments. The system integrates vision-based perception, task planning, and robotic manipulation to automate the book return process. Unlike traditional systems that rely on fixed coordinates, our system perceives the shelf in real-time, detects available gaps, and makes placement decisions based on current conditions.

The system workflow includes barcode-based book identification, RGB-D shelf perception, intelligent slot selection, and visual-servo placement control. After a single user input, the system processes batches of returned books without manual intervention.

### Individual Responsibilities and Role

My role in Team 34 focuses on hardware integration, control algorithms, and project coordination. I am responsible for the software-hardware interface, control implementation for the manipulation subsystem, and coordinating timelines across the team.

Specifically, I handle hardware selection and assembly, control system design, ESP32-ROS2 integration, project documentation, and milestone tracking. I also coordinate between the mechanical and electrical/software components.

## Individual Design Work

### Hardware Architecture and Component Selection

I designed the hardware architecture for the manipulation subsystem. The system uses a 6-DOF aluminum frame robotic arm with ZX15D serial bus servos. I selected and tested the servos, checking torque requirements, communication protocols, and compatibility with our control framework. The ZX15D servos provide 20 kg·cm torque at 7.4V with position feedback through serial bus communication.

The control architecture uses a laptop running Ubuntu 22.04 with ROS2 Humble, connected via USB to an ESP32 microcontroller that drives the servo bus. I tested the manufacturer's drivers through the ESP32 and verified Arduino compatibility. Hardware tests confirmed that servos operate within the required motion range, precision, and load capacity.

### Control System Implementation

I designed a ROS2-based control system using standard inverse kinematics packages and motion planning frameworks. The control pipeline uses OpenMV packages for kinematics, ROS2 MoveIt for trajectory planning, and real-time joint state feedback.

I implemented ESP32-ROS2 integration through rosserial communication, allowing the ESP32

to function as a ROS2 node that publishes joint states and receives motion commands. This enables the upper-level planner to receive servo feedback and send trajectory commands to the hardware. The basic communication pipeline is working, though integration with high-level decision algorithms is still in progress.

## **Physical Prototype Assembly**

I assembled the physical prototype, including the aluminum frame structure, six servo motors, ESP32 control board, and power distribution system. Assembly required mechanical integration of the arm structure, electrical wiring for the serial bus servo chain, and verification of workspace coverage relative to bookshelf dimensions. The assembled prototype can reach across the target shelf area.

## **Project Documentation and Coordination**

I wrote the Feasibility Analysis Report to standardize technical interfaces and component specifications across subsystems. This document was not required by the course but helped maintain alignment within the team.

For all major project documents (Project Proposal, Design Document), I wrote the manipulation and control subsystem sections and managed document structure, formatting, and submission. I set up document templates, coordinated content from all team members, and ensured consistent technical terminology and formatting.

I organize weekly team meetings to sync progress across subsystems. Over the past eight weeks, I have led progress reviews, identified integration dependencies, and assigned tasks for upcoming deadlines. I communicate milestone timelines 1-2 weeks in advance.

## **Testing and Verification Plans**

Current verification focuses on hardware-software integration and control algorithm validation. Planned tests include:

ESP32-ROS2 communication testing to validate message transmission, measure latency, and verify joint state feedback accuracy.

Inverse kinematics accuracy testing to compare commanded positions with measured positions using camera feedback.

Gripper force feedback validation to test stall detection through servo current and load monitoring via the JointState effort field.

End-to-end placement testing to integrate perception, planning, and manipulation modules for complete pick-and-place cycles.

Contingency plans include using open-loop trajectories if visual-servo integration has timing issues, and simplified gripper designs if force feedback is unreliable.

## **Conclusion**

### **Self-Assessment**

I estimate my contribution at approximately 30% of the total project workload. This includes

hardware selection and assembly, control system architecture, ESP32-ROS2 integration, project coordination, and documentation. My role as team organizer adds scheduling and communication overhead beyond technical implementation.

Current progress is on schedule. Hardware assembly and initial control integration are complete. The tight project schedule requires continued coordination to stay on track.

### **Plans for Remaining Work**

Current Focus: Complete hardware-software integration by finalizing ROS2 communication interfaces and validating inverse kinematics accuracy. Implement gripper force feedback through servo load monitoring. Begin integration with perception module outputs.

Future Stage 1: Test manipulation subsystem independently, measuring placement accuracy and gripper reliability. Calibrate camera-arm transformation for visual-servo control.

Future Stage 2: Run end-to-end system validation, integrating perception, planning, and manipulation subsystems. Measure placement success rate and adjust control parameters to meet target performance.

The goal is to achieve 90% placement success rate on the demonstration scenario. Additional objectives include adaptive gripper force control and visual-servo refinement for better placement stability.