Weekly Report
Week 3

Nguyen Huu Huy Thinh

2025-02-17

Abstract

This report summarizes the tasks completed, challenges faced, and progress made during the past week. Additionally, it outlines the next steps and any issues that require further discussion.

1 Introduction

During the past week, I have completed the robot arm, fixed the Arduino circuit, and implemented a simple interface to control the robot arm. In addition, I have been researching information related to the OAK-1 camera and working on creating a suitable environment for simulating plastering work.

2 Using Mastercam for designing robot arm

Mastercam is a powerful CAD/CAM software used for designing and machining parts. For the robot arm project, Mastercam has been essential in creating the design files needed for 3D printing and part fabrication.

Step 1: Setting up the workspaces

- Set up the analyze units to either millimeters or inches based on the 3D printer's requirements in Configurations -> Analyze.

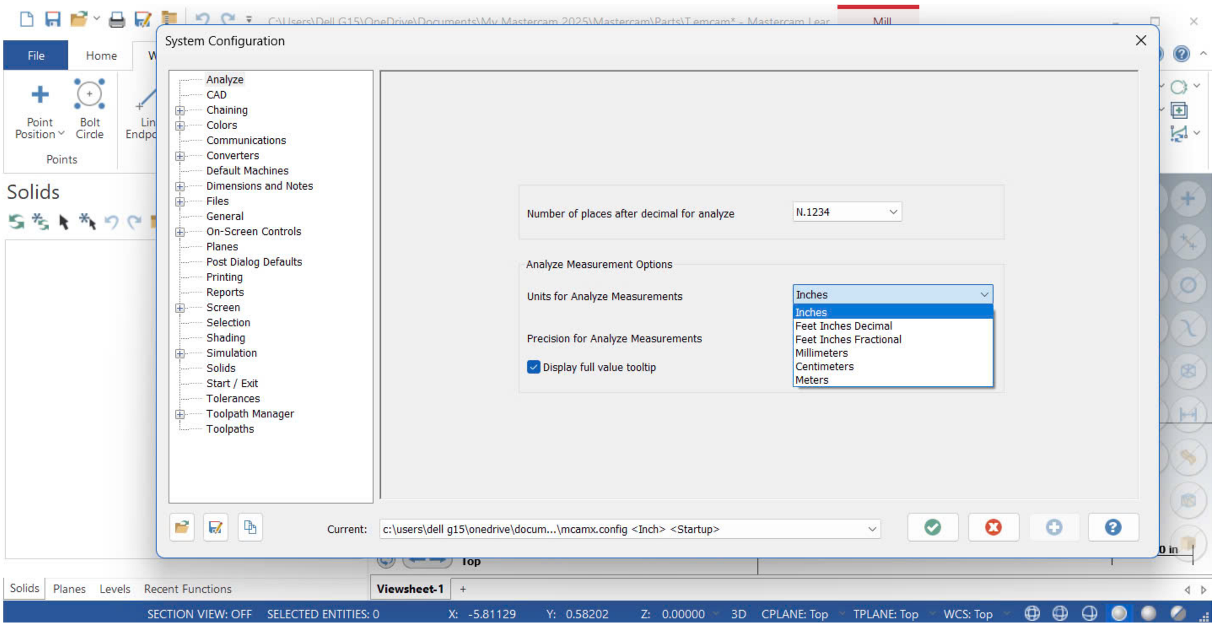

Figure 1: Setting up the workspaces

Step 2: Sketch the Base Shape

Begin by sketching the basic design of the robot arm components on the Oxy coordinate plane. Use the tools available in the Wireframe menu, such as Rectangle, Line, and Circle, to create the foundational shapes of the parts in a 2D area.

- Use trim to refine the shape
- Apply constraints and relations to maintain geometric accuracy.

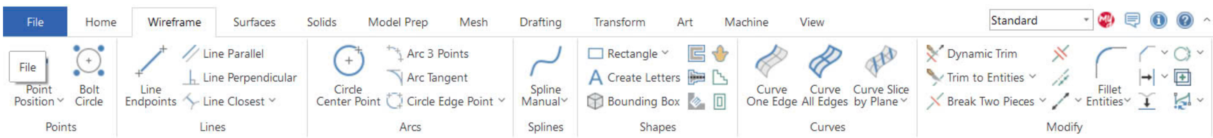

Figure 2: Wireframe Menu

Step 3: Extrude the Shape

Once the base shape is sketched, use the Extrude tool to turn the 2D design into a 3D object. This step defines the thickness and volume of the parts, creating the final solid model of the robot arm components that can be exported for 3D printing.

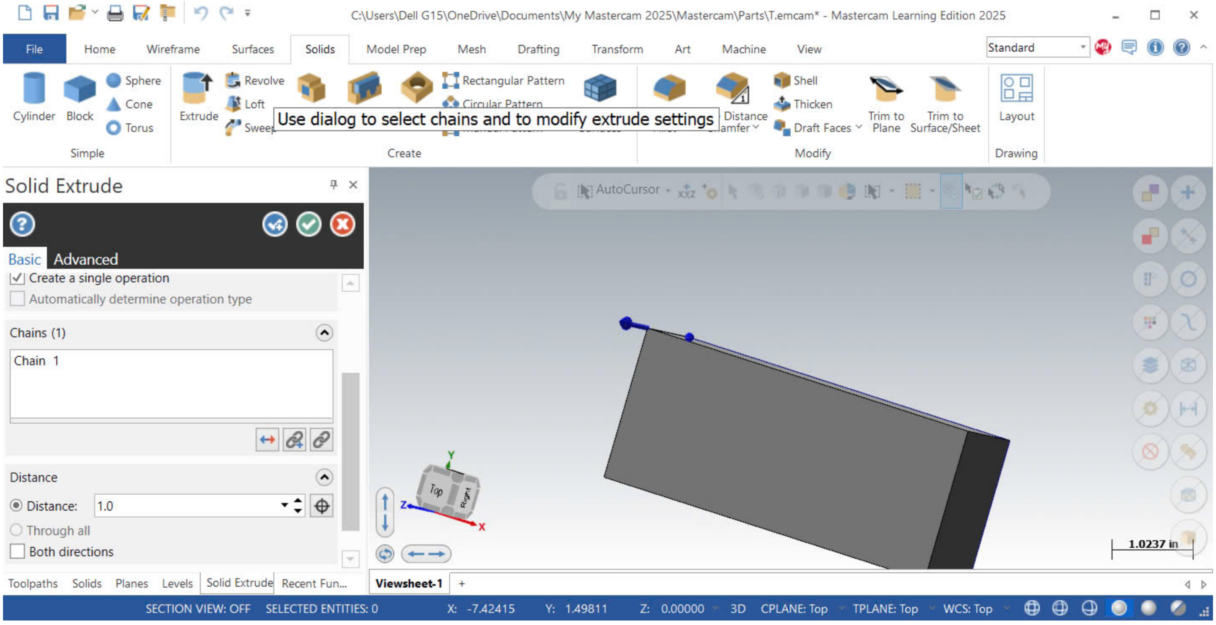

Figure 3: Extrude the shape using Solids Menu

This workflow in Mastercam is the basic process for creating a 3D model, which can then be exported for 3D printing or further machining.

3 Controlling Robot Arms

This week, I completed the design of the trowel attachment and successfully assembled it onto the robot arm.

Figure 4: 3D Printing Robot Arm

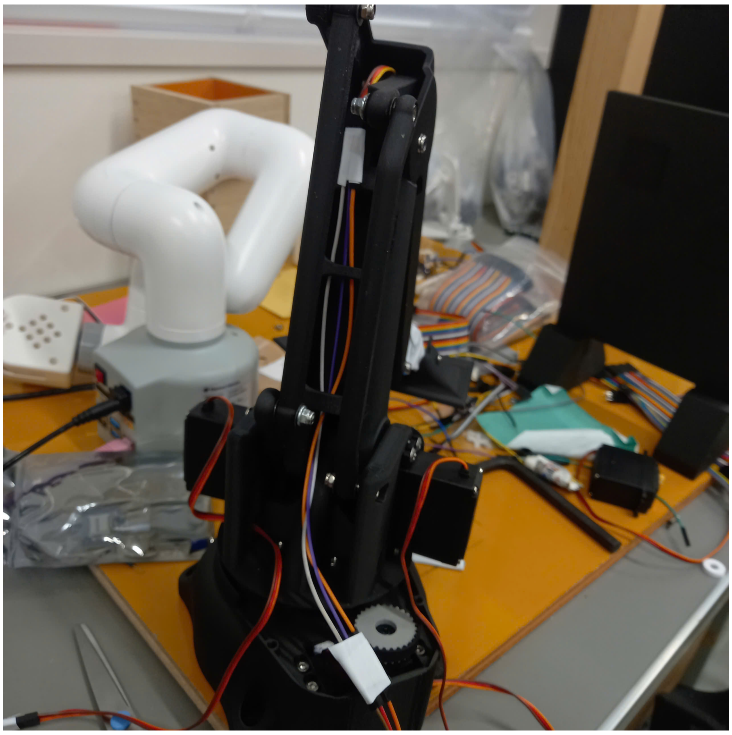

Figure 5: Trowel's Jumper Wire

Additionally, I connected the necessary wiring to the Arduino circuit, setting up the control system for the robot arm. This step is crucial for enabling precise movement and task execution. The connections were tested, and initial checks indicate proper communication between the Arduino and the servo motors.

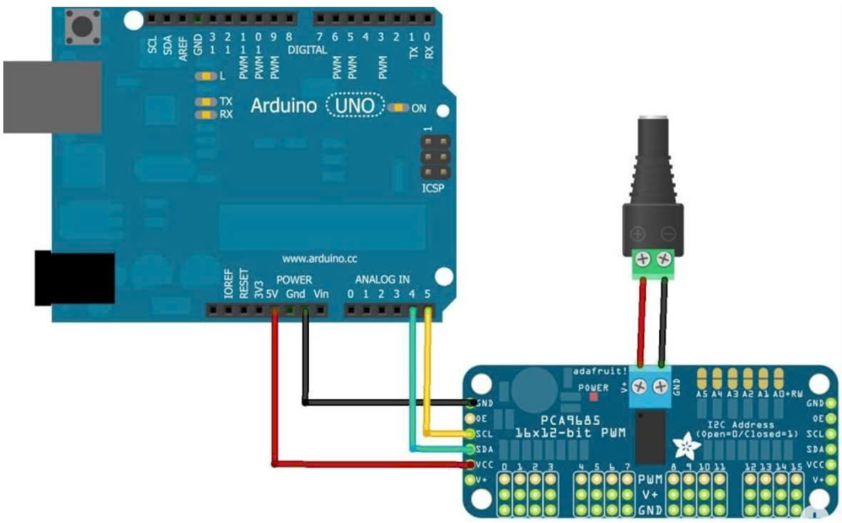

Figure 6: Arduino circuit for controlling robot arm

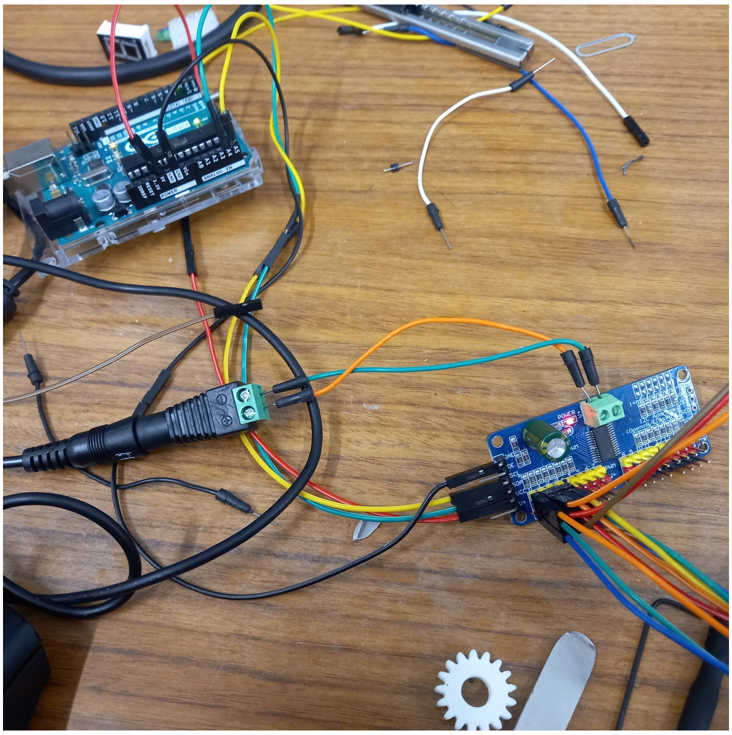

Figure 7: Arduino circuit for controlling robot arm

I also implemented a simple user interface for easy control of robot arms using PySerial and the Qt framework. The interface allows for smooth communication and command execution, improving usability and interaction with the hardware.

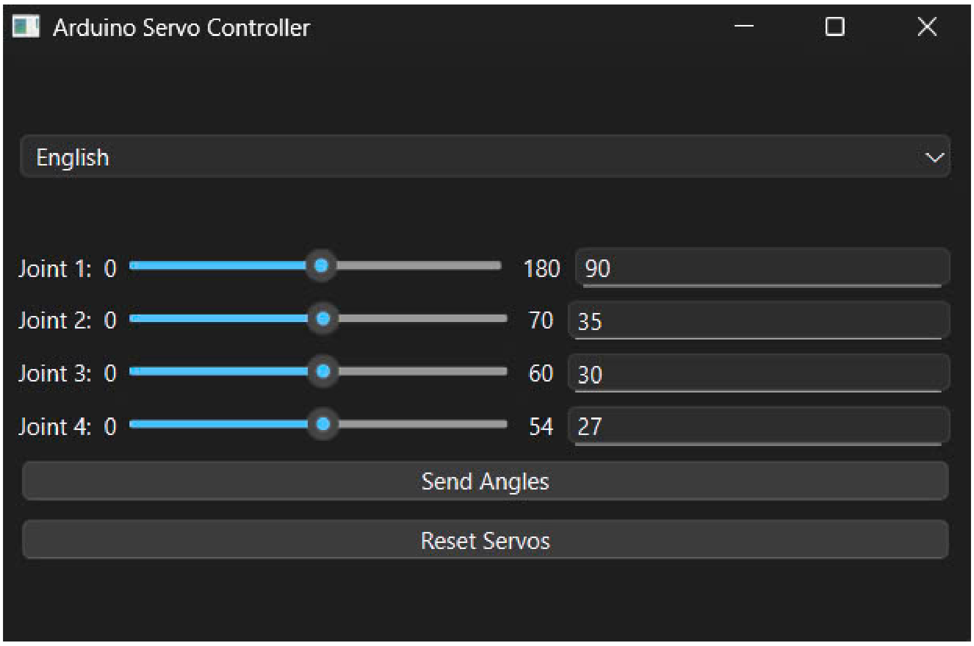

Figure 8: User Interface

Here's a structured description of the joint limits and rotation axes:

- Joint 1: Rotates around the Z-axis with an angle range of $-45° \leq x \leq 45°$. It is connected to the motor through a pulley structure, allowing controlled motion transmission.

- Joint 2: Rotates around the X-axis (Ox) or Y-axis (Oy), depending on the configuration, with an angle range of $0° \leq x \leq 70°$. It has a 1:1 motor-to-joint ratio for precise control.
- Joint 3: Rotates around the X-axis (Ox) or Y-axis (Oy) as well, with an angle range of $0° \leq x \leq 60°$, with a 1:1 motor-to-joint connection for direct rotation.
- Joint 4: Rotates around the X-axis (Ox) or Y-axis (Oy) as well, with an angle range of $0° \leq x \leq 54°$, having a 1:1 motor-to-joint connection for smooth movement.

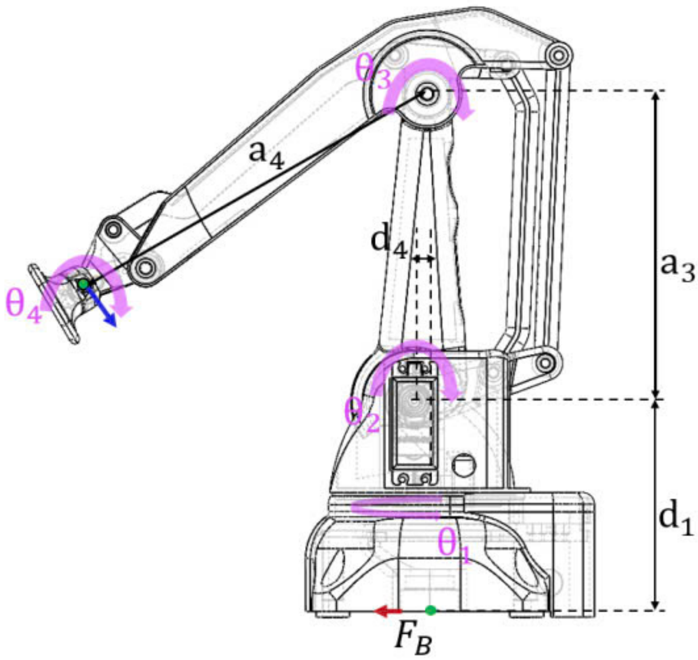

Figure 9: Robot Arm Design [1]

4 Setting Up Environment for Plastering

First, I researched how to connect the OAK-1 camera to the workstation and learned how to use DepthAI to integrate an AI model for segmentation into the camera. Since the robot design was not suitable for mounting the camera in an optimal position, I decided to create a custom base to fix the OAK-1 camera in place, ensuring it captures the entire wall. The wall will be 3D printed with removable shapes that can be rearranged to create various datasets. Additionally, the legs of the wall can be detached to allow the creation of different wall configurations with varying obstacles to ensure diversity in the dataset.

Figure 10: Setting Up Environment

Figure 11: Wall

5 Next Steps

- Create additional walls that can be used for plastering tasks.
- Collect data for fine-tuning the segmentation model and controlling the robot arms.

6 Conclusion

In conclusion, this week I focused on setting up the environment for plastering and making progress on the robot arm design. I researched the connection of the OAK-1 camera and learned to integrate an AI model for segmentation. Additionally, I addressed the issue of mounting the camera by designing a custom base. The 3D printed wall with removable shapes will allow for diverse datasets.

References

[1] S. Dong, M. Muramoto, and W. Du, "A custom robotic arm simulation using RoboDK for plastering work," in Proceedings of the 2024 10th International Conference on Computing and Artificial Intelligence (ICCAI '24), 2024.