

Team: Joshua Siderits, Matina Ypsilantis, Joshua Liang, Ryland Vick, Bryan Chang, Jared Mahotiere, Joshua Wilson

Mentors: Ainsley Phipps, Frederick Berry

## Customer Background

The School of Engineering Technology is the the sponsor for this project. This project has been a continuation from previous years in training robot arms to do specified tasks. This project has been lead by previous Capstone student, Ainsley Phipps and Capstone professor, Fred Berry.

## Problem Statement

Many industries still rely on manual labor for repetitive and dangerous manual tasks, which lead to a high rate of injuries. Traditional industrial robots are often expensive and lack two-handed flexibility. This project aims to offer a cheaper option by implementing reinforcement learning on a bimanual robot arm.

## Requirements Matrix

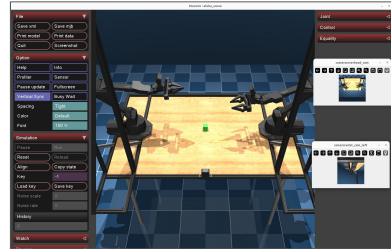
Req #	Design Requirements	Design Targets	Validation
	<b>Rationale</b>		
1	Multiple New Laptops	Laptop needs at least a 5090 processor and 22.5 vram	Need the new laptops to perform required tasks
	Without the New laptops we are unable to perform the necessary tasks		
2	Implement $\pi 0$	Use the given database to implement with our arms	With $\pi 0$ would allow for more efficient running of the arms
	With the better performance of the robot arms would allow for more accurate demos and improve arm capabilities		

## What is Reinforcement Learning?

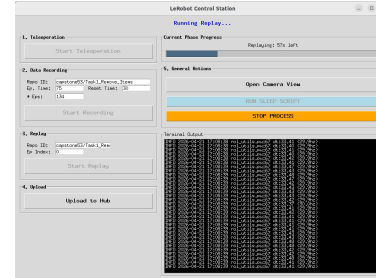
Reinforcement learning is a computer's version of trial and error. Through continuous tests and training, the computer can generate the "best fit" course of motor action. During training, the robot is given "Rewards" and "Penalties" that affects the robot's actions. Through incorporation of computer vision, the final product results in a somewhat adaptable summarization of all of the trials.

## Final Design

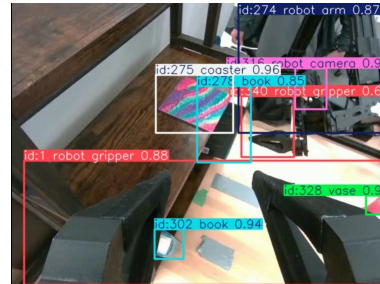
Simulation



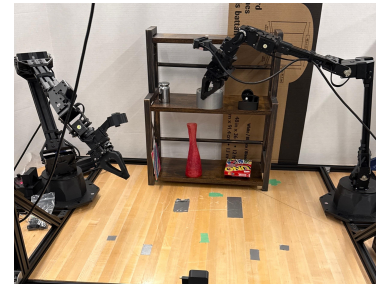
GUI



YOLO



Physical



## Physical Training

The physical training was done by using teleoperation to move the robot arms to complete tasks. On each run, episodes from all four camera perspectives were recorded and saved.

## Simulation

The robot arms were modeled using Mujoco, a basic physics simulator, and trained using reinforcement learning through the Gymnasium library. A multi-input reinforcement policy receiving feedback from the cameras and the positions of the arms regarding the items surrounding. Simulation is cost effective by using Linux and Visual Studio Code as well as time efficient as training can be done without human intervention.

## Object Detection

YOLO is a visual AI software tool to detect objects. It does not create a model automatically; it needs to be trained. In our project, its role is to train our Trossen robot arm system to find different objects from its camera perspectives, so the robot knows what to do with them.