Robotic Design for Strategic Adversarial Games

Kylie Herbstzuber (RBE), Elene Kajaia (RBE/CS), Luca Dang (CS/MA), Leo Riesenbach (ECE/CS), Mason Sakai (RBE), Dylan Schmit (RBE/CS), Vishesh Konduru (RBE), Graham Mack (RBE)

Advisors: Professor Griffin Tabor (RBE), Brad Miller (RBE), Professor Xinming Huang (ECE/CS)

Abstract

Worldwide, over 10,000 high school robotics teams compete in a fast-paced, annual competition using teleoperated robots. Our goal is to demonstrate that a fully autonomous robot can compete at this same level. Autonomous performance in dynamic, adversarial environments remains a major challenge in robotics, requiring systems that can continuously interpret the environment and respond in real time. This project aims to develop a fully autonomous system that integrates multi-sensor perception, real-time state estimation, and a strategic decision-making model to determine how to respond to an opponent. By translating advanced robotics into a familiar and engaging context, we aim to inspire the next generation of engineers.

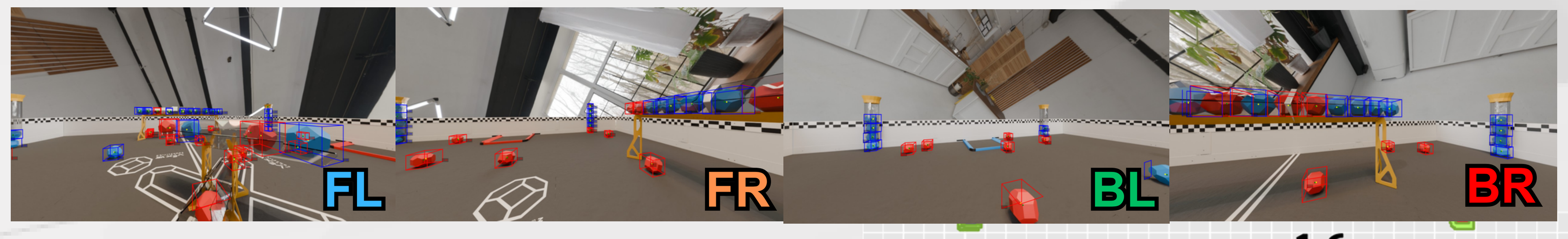
Vision + Object Detection

Real-World Camera View (Post-Processed)

Unified Neural Network for Robot Vision and Object Detection

- **Input**: 4-camera sensor array, **Output:** Positions of all game objects
- ~30M parameters, trained on 4 L40S (24h)
- Dataset: 200k synthetic scenes generated with Blender

Synthetic Data Example



Evaluation Parameters

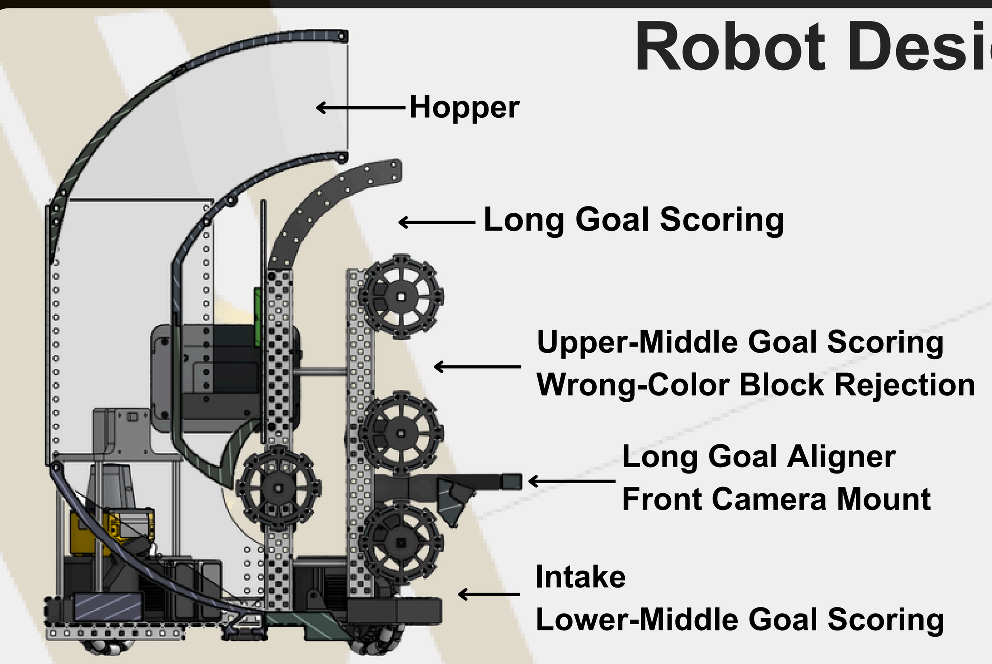
- Using greedy matching[1] technique
- Classification: Precision, Recall
- Localization: L2 Mean

Evaluation Results:
(Performance at varying object distances; P = Precision, R = Recall, L2 = L2 Mean)

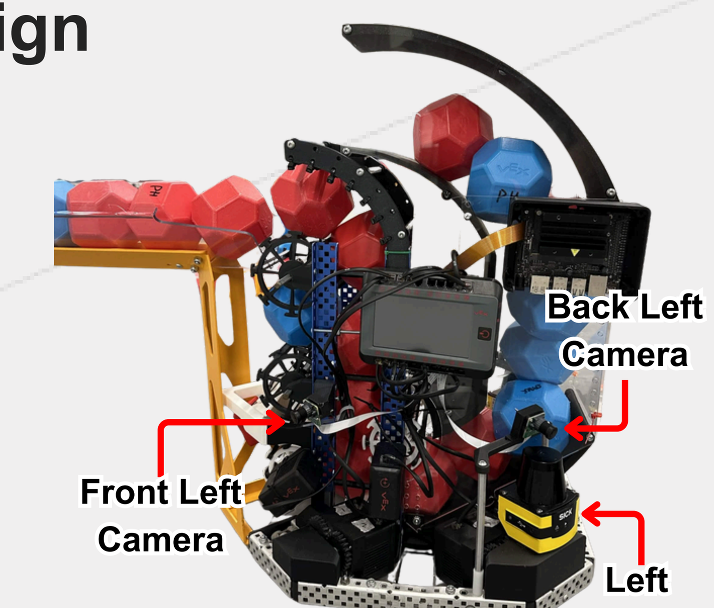
- **0.0–0.8 m → P:** 0.9118 **R:** 0.7487 **L2:** 0.93 cm
- **0.8–1.6 m → P:** 0.9164 **R:** 0.8215 **L2:** 0.91 cm
- **1.6–3.2 m → P:** 0.8813 **R:** 0.8130 **L2:** 0.98 cm
- **3.2–6.4 m → P:** 0.7894 **R:** 0.6115 **L2:** 1.09 cm

1. nuScenes (Caesar et al., 2020)

Robot Design

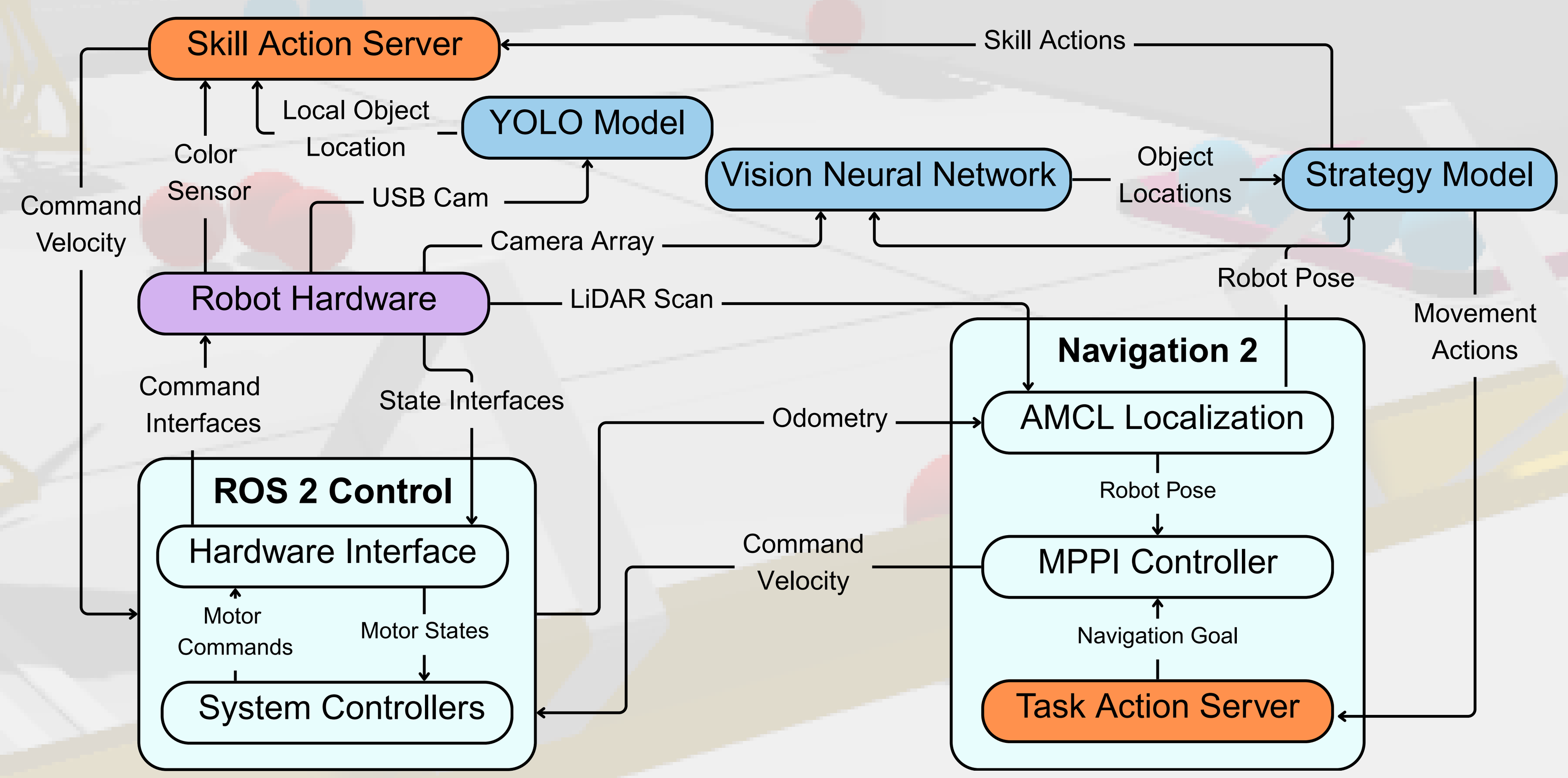


CAD Cross-Section View



Real-World Robot

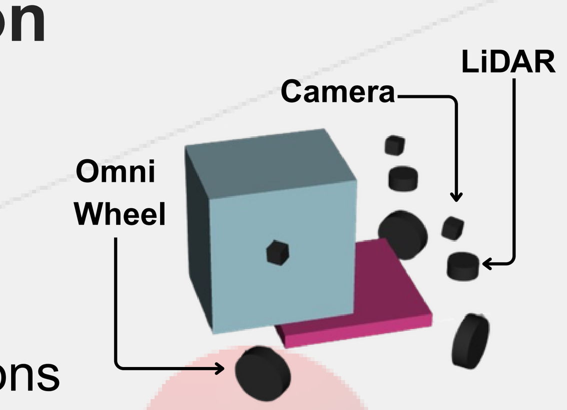
Software Architecture



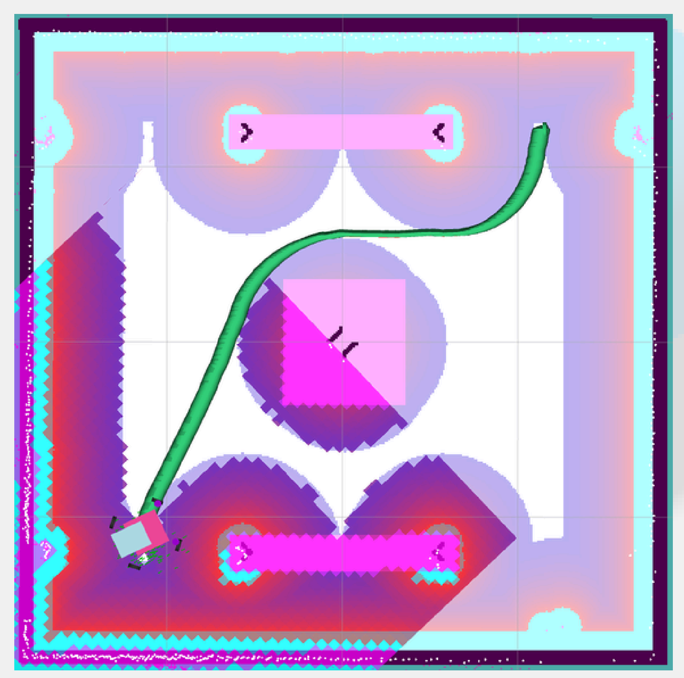
ROS 2 & Simulation

Core ROS 2 Stack

- Navigation 2 (Nav2)
  - Adaptive Monte-Carlo Localizer
  - Model Predictive Path Integral Controller
  - BehaviorTrees for regulating robot decisions
- ROS 2 Control
  - Omni Wheel Drive Controller used to control the robot base
  - Forward Velocity Commanders for each intake motor
  - Hardware interface for communicating with the VEX V5 Brain



Robot URDF for Simulation

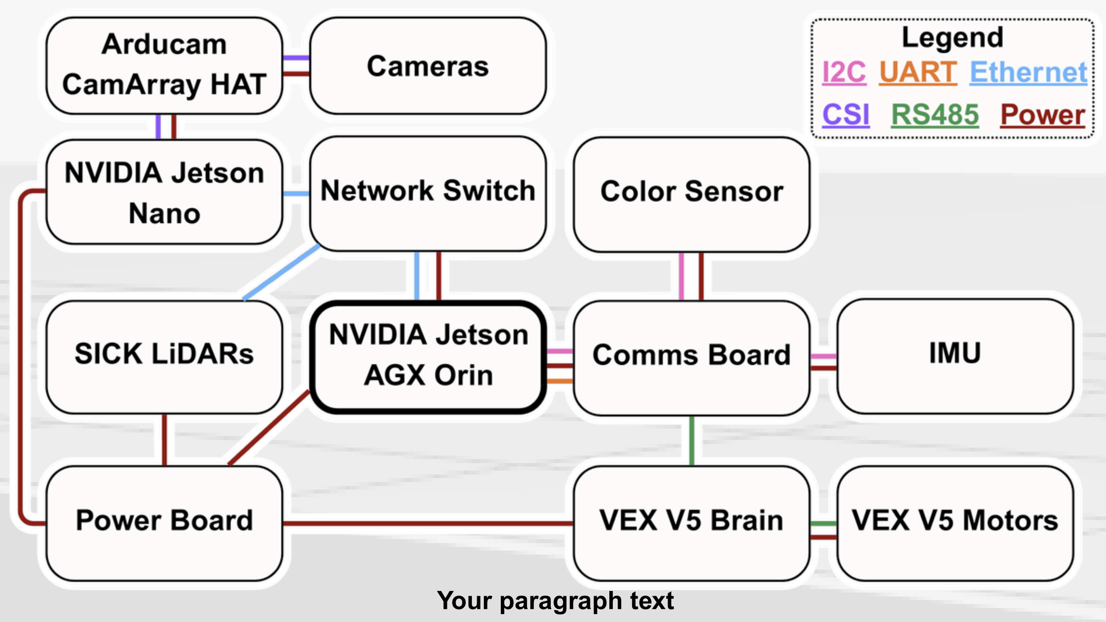


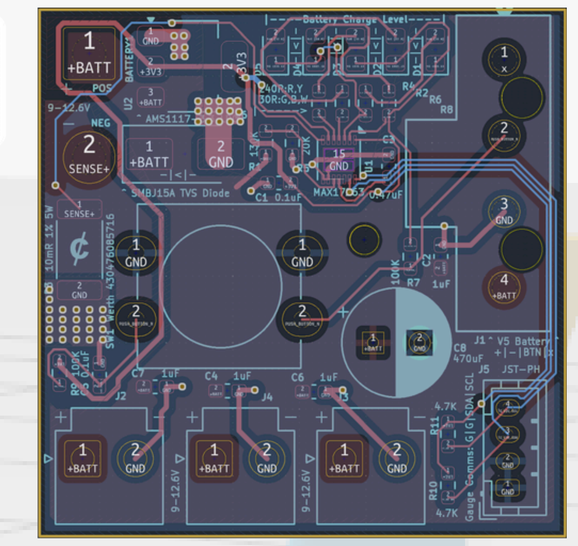
Visualizing the Nav2 MPPI Controller Path in Rviz

Developing in Simulation

- Built URDF robot model and tested core functionality in Gazebo Sim before deploying on hardware
- Able to play the full game in simulation
  - Services for controlling various field and game elements
  - Identified shortcomings in simulation and created a plugin to improve it

Electronics





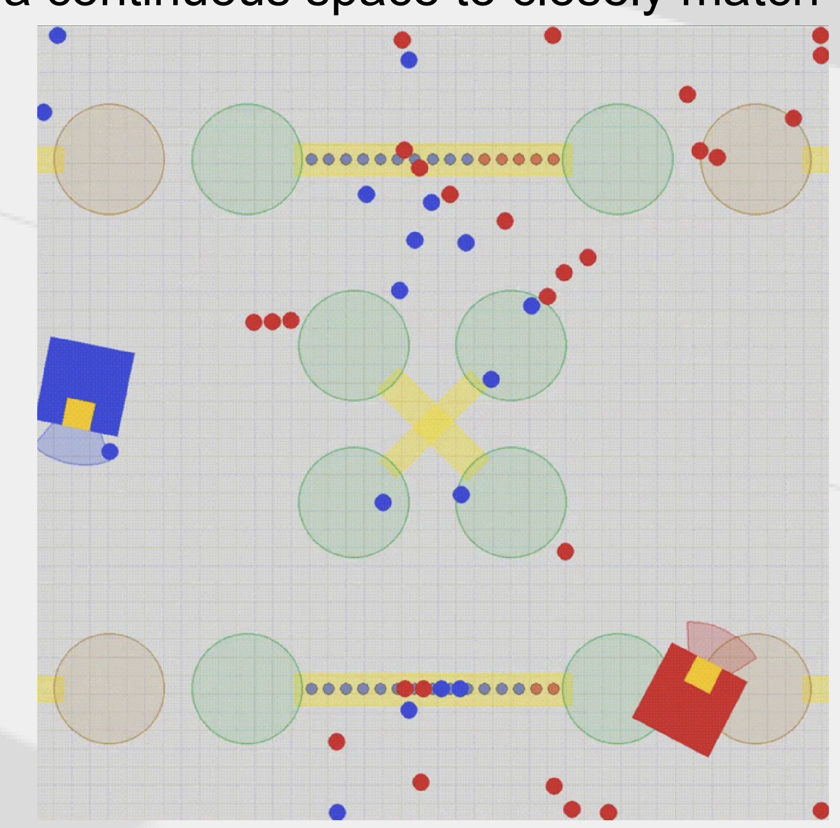
PCB Layout of Power Board

- NVIDIA Jetson AGX Orin runs the ROS 2 system and AI models
- Comms board adapts Jetson to I²C sensors and RS-485 actuators
- Power board distributes 12V to Jetson, LiDARS and VEX hardware
- Camera images are synchronized with CamArray and sent over Ethernet

Strategy Model

- **Goal**: take in the current state of the game and give out an action to play
- Trained using a simulated game in a continuous space to closely match the real world

Input: field state (robot, ball, goals, and loaders' locations, etc.)
Output: action state (pick-up, score, block, move)



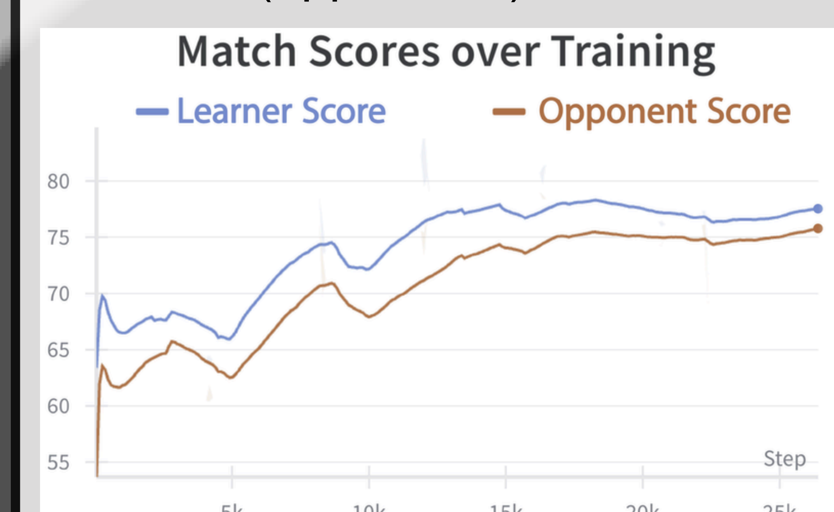
Visualization of Training Environment

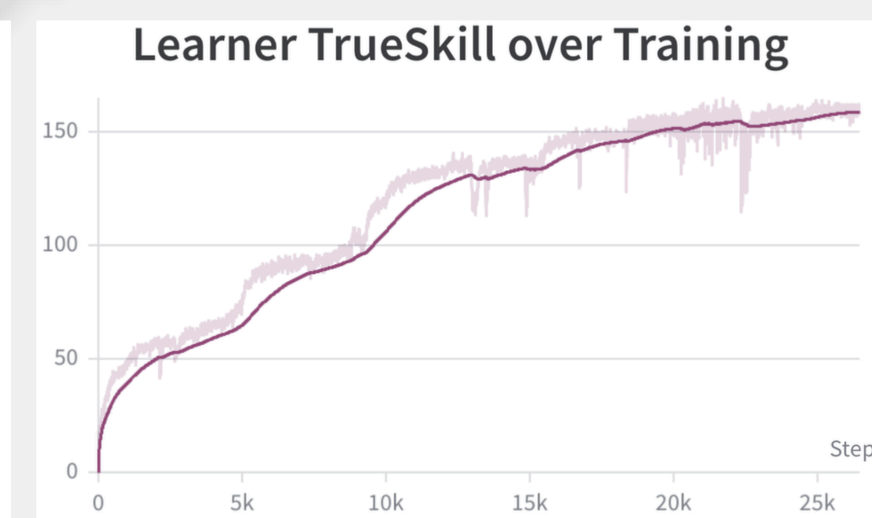
Architecture and Training

Pipeline: ~1.5M parameter MLP model trained with reinforcement learning over 150 hours for ~26M games

Training and Evaluation

- Model snapshots were evaluated and scored with TrueSkill
- The current model (learner) version would play old snapshots of the model (opponents) to evaluate the current model





Berner, C., Brockman, G., Chan, B., Cheung, V., Dębiak, P., Dennison, C., Farhi, D., Fischer, Q., Hashme, S., Hesse, C., Józefowicz, R., Gray, S., Olsson, C., Pachocki, J., Petrov, M., Pinto, H. P., Raiman, J., Salimans, T., Schlatter, J., . . . Zhang, S. (2019). Dota 2 with Large Scale Deep Reinforcement Learning. ArXiv. https://arxiv.org/abs/1912.06680