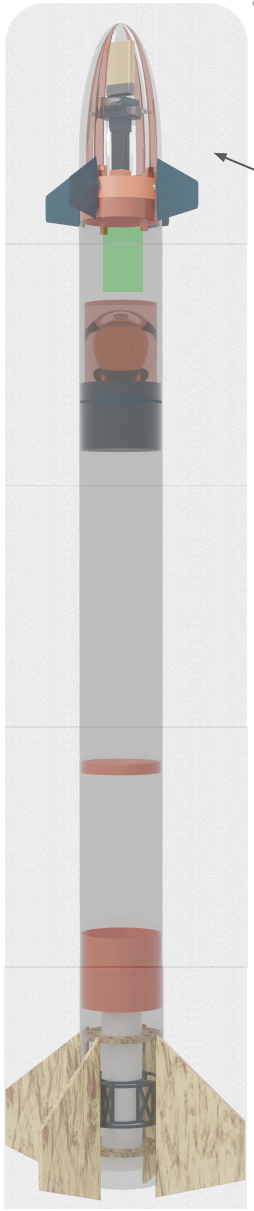
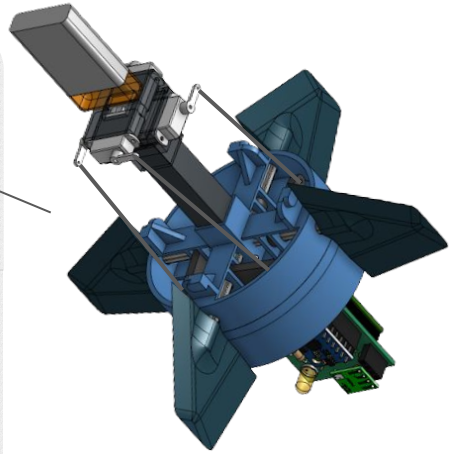
Rohan Bandaru Maker Portfolio

(https://www.linkedin.com/in/rohan-bandaru-a59270166/)
(rohanbandaru1@gmail.com)

Personal site: https://rohanbandaru.github.io/site/

Model Rocket Guidance System - Mechanical



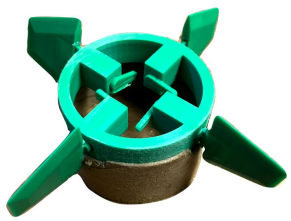


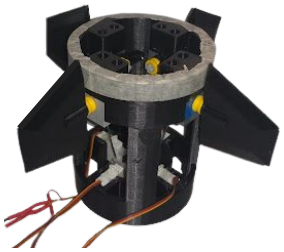
Control Actuator System (CAS)

I used CAD and 3D printing, CNC Milling, to fabricate parts.

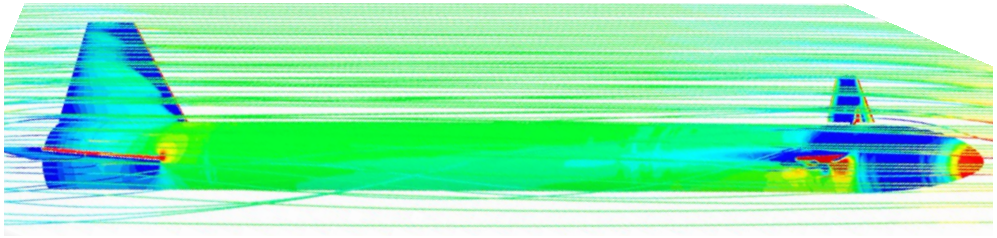
Calculated optimal linkage geometry, maximizing torque at desired reaction speed throughout the range of motion. Using linkages allowed for lighter servos and battery whilst protecting them from breaking.

Placing canard pivots above center of pressure saves the vehicle in the event of GNC failure (canards turn into airstream).

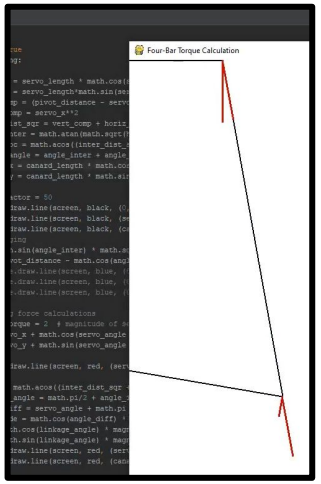




Multiple revisions, optimizing for weight, torque, and canard geometry.



CFD to characterize canard torque response, ensure no control reversal. Used Openrocket to achieve desired aerodynamic stability.



Python GUI for visualizing linkage geometries

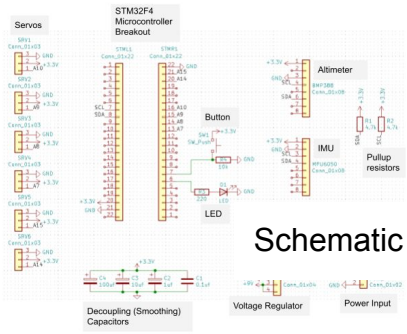
Model Rocket Guidance System - Electrical

State indication Button Inputs

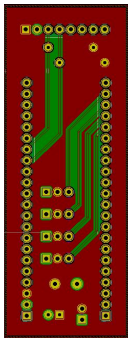
VREG 6-Axis IMU Barometer

I designed a custom PCB (Printed Circuit Board) in KiCAD

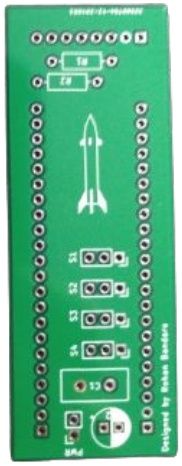
- STM32 Arm Microcontroller @84MHz
- 6-Axis IMU (Gyro + Accelerometer) for Orientation + Position
- Barometer (+Temperature, Humidity) for Altitude
- Flash Chip for data logging
- LEDs + Buzzer* for state indication
- Multiple button inputs for adjustment on the pad

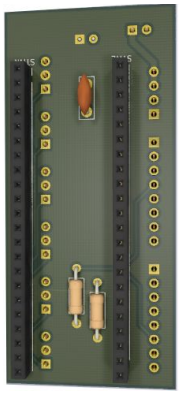


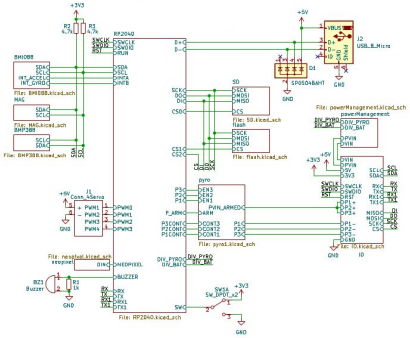
Schematic



Layout

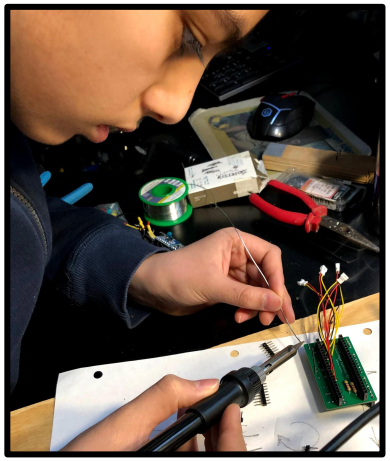






PCB

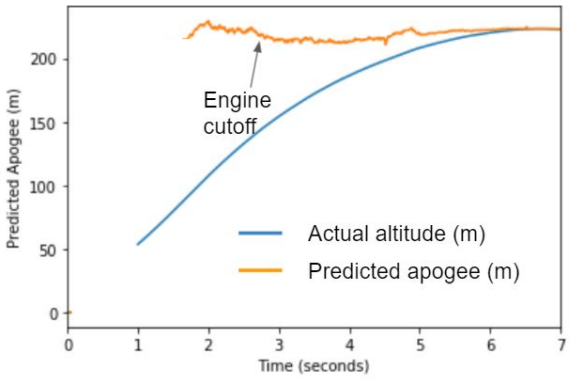
Board went through many revisions, adding features and making it compact. (Surface-Mount version not finished due to chip shortage and cost 😕)



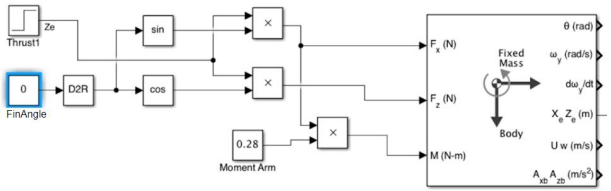
Utilized Through-hole technology (THT) since I didn't have hot-air station or reflow oven. Used pin headers with breakout boards, making them swappable for repairs or upgrades, and letting me put components underneath.

Chose a 1S 600mAh LiPo to minimize weight while preserving ~40 minute run time, depending on servo movement. Allowed for prolonged pad idle and recovery.

Model Rocket Guidance System - Software



Real-time Apogee prediction using kinematic model



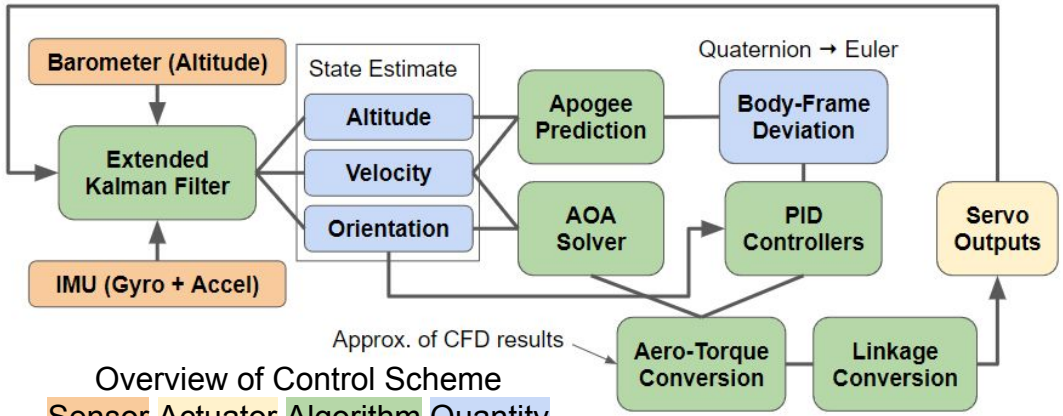
Simulink Simulation
(found to be inaccurate due to aerodynamics)

Wrote firmware in Arduino using Platformio and STM32duino
Arduino Core adds minimal overhead, while bringing a lot of functionality/support and I had previous experience using it.

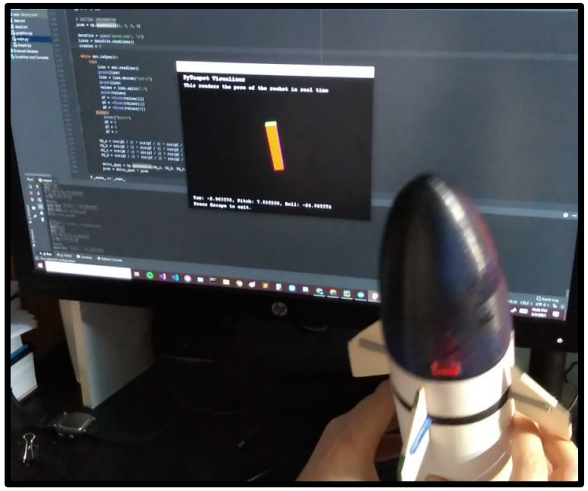
Implemented an Extended Kalman Filter for pose estimation. It combines the short-term precision of the IMU with the long-term accuracy of the barometer to produce a good altitude estimate. The other axes suffer from drift, but they are not important for apogee correction.

Used Quaternions for tracking orientation/rotation, as they are computationally efficient. Converted back to Euler angles for correction.

PID controllers for each body-frame axis (roll, pitch, yaw) output a desired torque to converge to target orientation. Canard deflection imparts variable torque on the vehicle, depending on Angle of Attack (AOA) and airspeed, so that is estimated for each canard, and a servo PWM signal is produced.



Overview of Control Scheme
Sensor Actuator Algorithm Quantity



Testing orientation tracking tethered to my computer

*System is incapable of hitting point targets and is allowed under FAA rules

Model Rocket Guidance - Overview

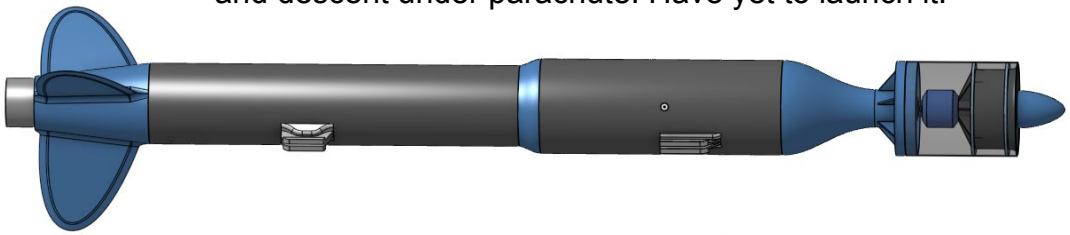
I worked on the project from September 2021 - June 2022, and presented it at the 2022 American Junior Academy of Science conference. This project taught me a lot about control theory, robotics, and aerospace.

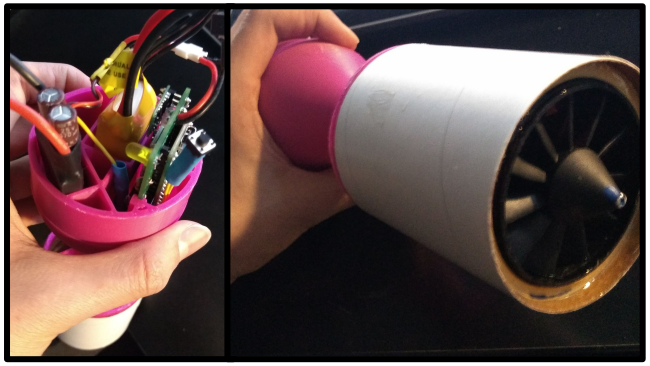
1st Place MIT MAssJAS Symposium 2021
1st Place MSEF (MA Science Fair) 2021
TARC National Finalists (with team) 2021

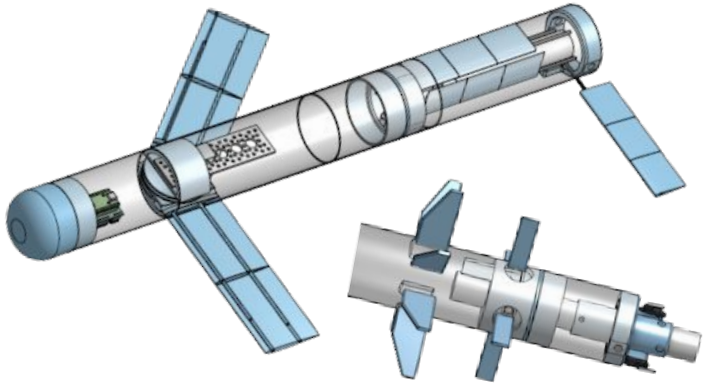
Team Rocket Projects:

In my Junior year, after co-founding the LHS Aerospace Club, I brought on teammates to continue exploring advanced model rocketry. Below are some cool/unconventional experiments we did for fun:

EDF (Electric Ducted Fan) nose cone for correction on ascent and descent under parachute. Have yet to launch it.







Deployable wings for tube-launched glide rocket, and thrust-vector control (TVC) Mount. Still in design phase.

Innovative airbrake design, using parts of the fuselage as drag surfaces. Ended up working very well.

PYPOSE

To connect classic robotics with modern learning methods seamlessly.

https://pypose.org/

Submitted ICRA (IEEE) 2022
https://arxiv.org/abs/2209.15428

Dates: Aug. 2021 - June 2022

Team: CMU AIRLAB

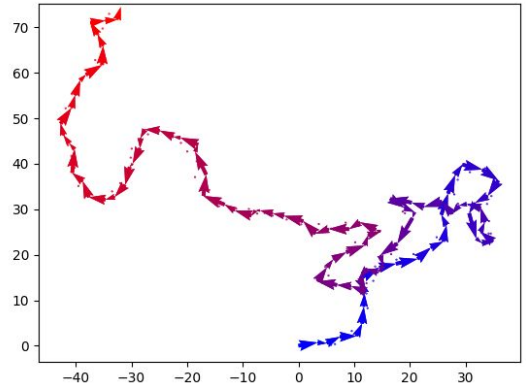
PyPose is a PyTorch-based Python library for Robot Learning with Physics-based Optimization. I completed this project as an intern at the Carnegie Mellon AirLab under the mentorship of Dr. Chen Wang.

Main features are:

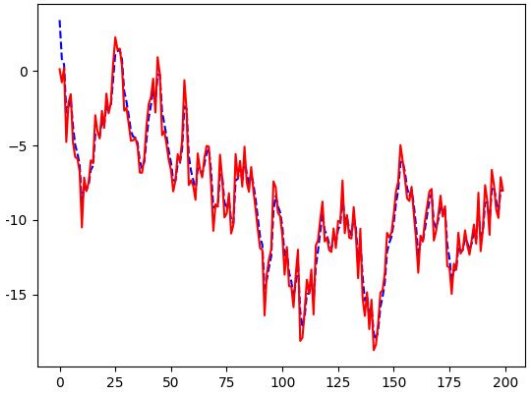
- Fast computation of Lie Groups (for pose transformation)
- Pose Graph Optimization (PGO)
- Dynamic System Models
- State Estimation (Kalman Filters)
- Controllers (PID, LQR, MPC)

My additions enable integration of Extended+Unscented Kalman Filters, System Models, and Controllers into PyTorch machine learning models

This project taught me a lot in Linear Algebra, Multivariable Calculus, Optimal State Estimation Techniques, Technical Documentation, Git VCS, and collaboration in an academic environment.



Example Tank Robot dynamic system. As robot traverses 2D space, pose estimate is propagated using EKF



Blue is filtered estimate.
Red is measured state variable.
Initial deviation is converged.

Robots like this AirLab drone can use PyPose to incorporate machine learning into pose-estimation.

Logicodex Library @ MITRE Corporation

Dates: June - Aug. 2022
Mentor: Adam Woodbury, Chief Technical Engineer, Embedded Systems, MITRE

Over the summer of 2022, I was an Embedded Security Technical Aide at MITRE.

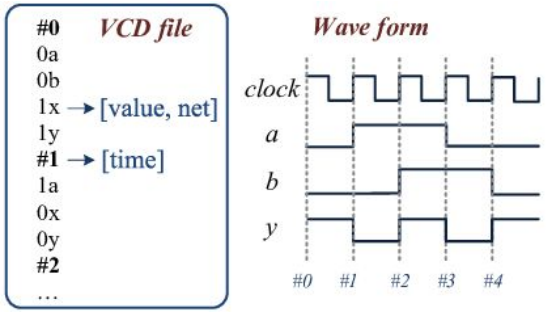
Created a target embedded system for testing Logic Analyzers (LAs), with modular C firmware for different protocols. Due to the sensitive nature of MITRE's work, I can't show pictures.

Created the Logicodex Python library to improve Logic Analyzer interoperability. Logicodex provides a simple interface for 4 LAs, ranging in price from $14 to $200k, allowing users to port data, synchronize captures, manipulate channels, and more. I wrote technical documentation for the LAs, enabling Embedded Systems engineers to rapidly switch between them, allowing for better use of tools.

Wrote memory-efficient and maintainable Python code, as the data was sometimes gigabytes long, and streamed at high speeds. I gained many skills in embedded Systems

Currently Logicodex is MITRE IP, but it's slated for public release.

Learned to use Keysight U4164a Logic Analyzers



Logicodex uses the Verilog Value-Change-Dump standard internally

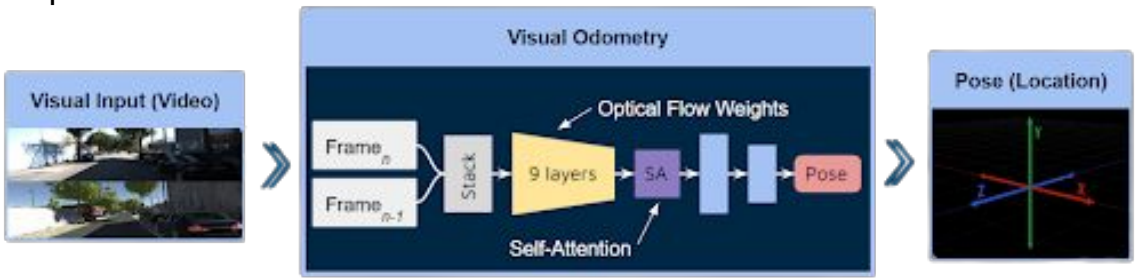
I reversed engineered Saleae's proprietary binaries

Deep Visual Odometry

Dates: Sep. 2021-May 2022
Team: Parth Kocheta

🏆 Grand Prize (1st Place)
MSEF (MA Science Fair) 2022

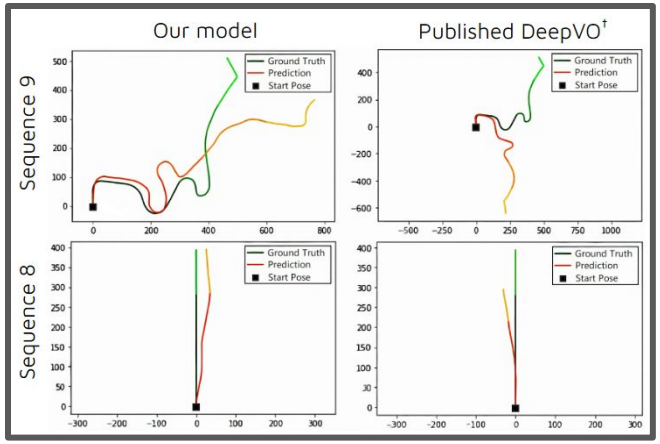
I had long been interested in the SLAM (Simultaneous Localization And Mapping) problem, and theorized a fully learned Visual SLAM policy could be more robust than classical methods. We first focused on the localization component (Odometry). Visual Odometry is a popular technique for robot localization, using cameras on the robot to track position.



Pytorch to create the models and AWS to train them. Supplemented our real robot data with Unity simulations and published datasets

After literature review, we created a novel model architecture, and tested it in various conditions against other approaches. Although it didn't outperform classical feature-based methods, we saw improvements over previous learned algorithms, and increased reliability in feature-scarce environments.

Test robot environment in my basement



Sample KITTI trajectories

FTC Robotics

Dates: 2017-2021
Team: 14039 Irrational

🏆 MA & NJ State Champs 2020
#1 OPR in the world 2020
#5 Autonomous World Rank 2020
MA State Finalists 2019, 2018

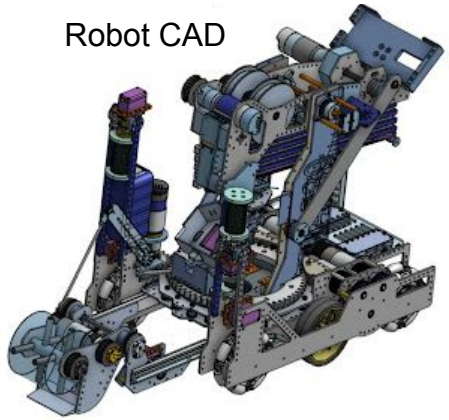
I was the lead programmer for FIRST Tech Challenge team 14039 IrRaTiONAl (Irrational).

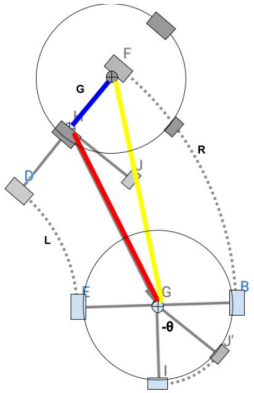
As my skills progressed, I implemented more complex algorithms onto our robot, such as Tracking Wheel Odometry, PIDF Control, Pure Pursuit, Jerk-Minimized Motion Profiling, Computer Vision Object Detection, and more recently, an Extended Kalman Filter for localization.

In 2019-2020 season, we had the 5th highest autonomous ranking out of 7000+ teams, and the number one OPR (Offensive Power Rating) in the world. (See ftcstats.org)

I'm proud to have derived and programmed all of our localization/control algorithms, instead of opting for the widely used RoadRunner plug & play library.

Robot CAD





Derivation of wheeled odometry

Students with a passion for robotics

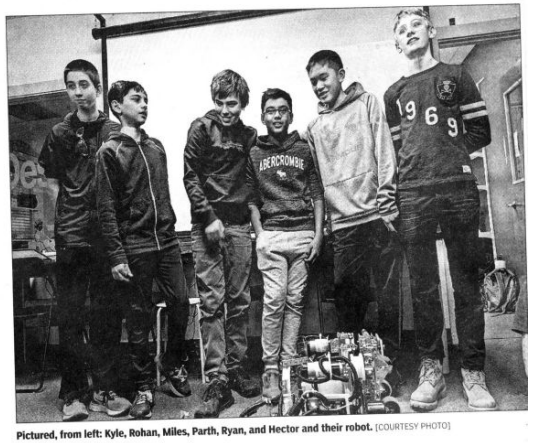
Diamond Middle School Robotics team advance to state championship

By Rohan Bandaru
Submitted to the Minuteman

75, 80, the points kept racking up.

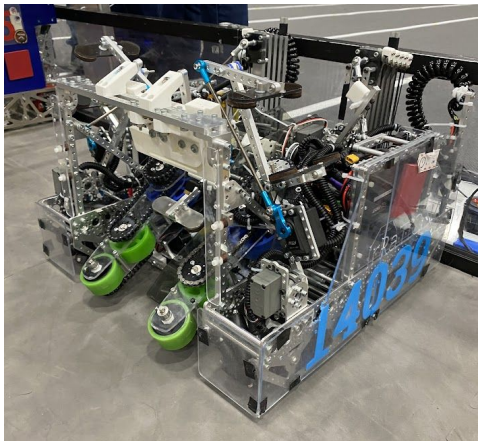
A ball teetered on the edge of the flipper. Slowly it rolled off the plate and landed right among the gears. There was a slight grinding as the plate got caught against the ball. Looks of horror slowly spread across our faces. None of the servos were responding, our robot was incapacitated.

Ryan and Miles, our drivers, desperately tried everything they could. Our robot had lost all ability to pick up any more minerals (balls and cubes) and deliver them to the lander. It could barely limp across the field. Our drivers tried to corral some minerals into the depot, but they were quickly swept away by the opposing team. This was the finals match of the qualifying round, held at Canton High



Pictured, from left: Kyle, Rohan, Miles, Parth, Ryan, and Hector and their robot. [COURTESY PHOTO]

event and won the Rockwell



2020 Robot

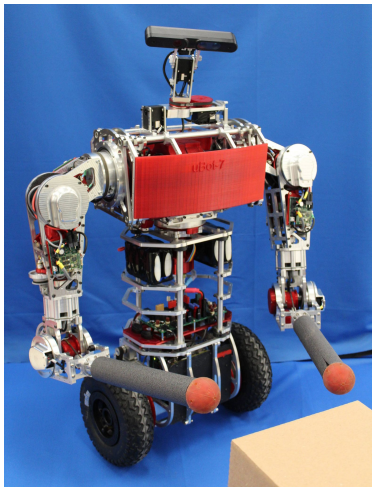
RL Control Hierarchy

Dates: May - Oct. 2020
Team: UMass Amherst Laboratory for Perceptual Robotics (LPR).

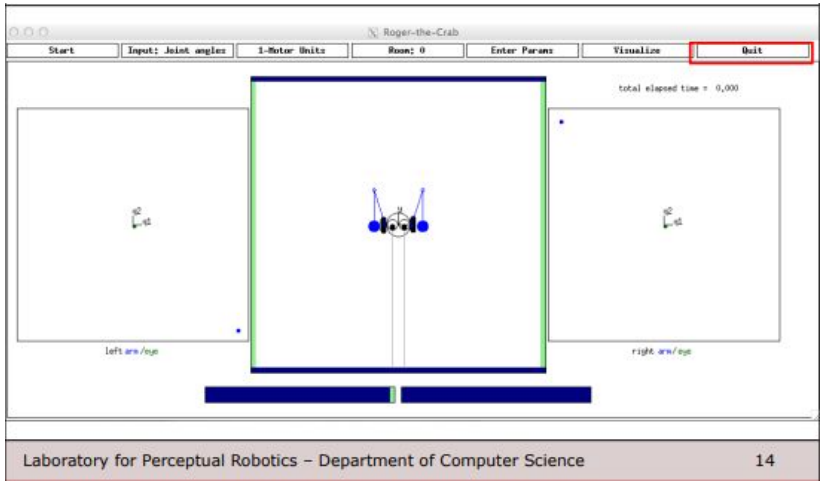
Worked with 2 other grad students under Prof. Rod Grupen on a series of experiments using the “Roger the Crab” simulator. I developed simulator tools, modernizing the old C codebase from the 90s, adding Python utilities for creating objects and environments. These experiments led to research (ICDL 2022) for a novel reinforcement learning control hierarchy, implemented in the simulator.

Created a “room painter” tool, that allowed users to generate new environments through a GUI, or automatically through scripts.

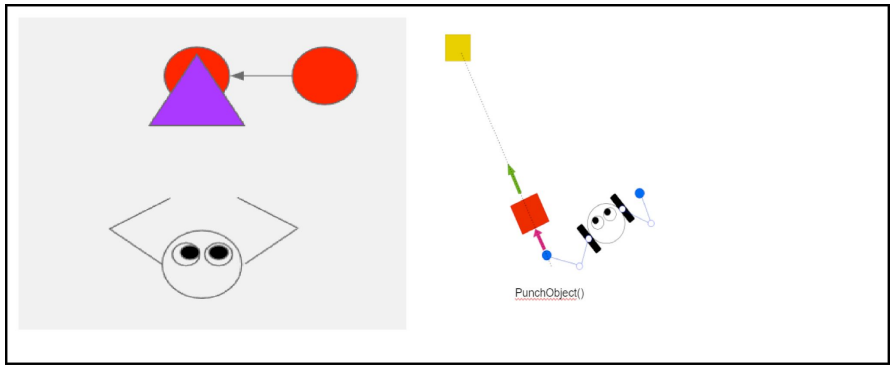
Cleaned up the collision/occupancy grid code to allow new objects to be added (Triangles, Squares, of different properties)



uBot 7 - a real life analog to Roger the Crab

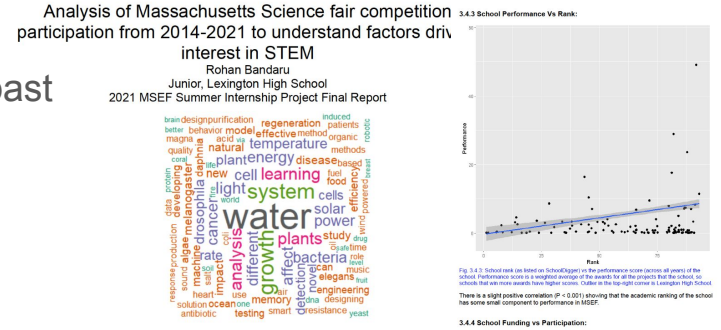


Screenshot of the simulator

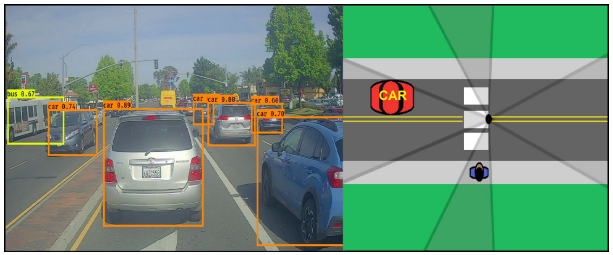


Example sketches from experiment discussion
Left: Object Permanence, Right: Motion Prediction

Other Projects (See Website: https://rohan2017.github.io/rohanbandaru/)

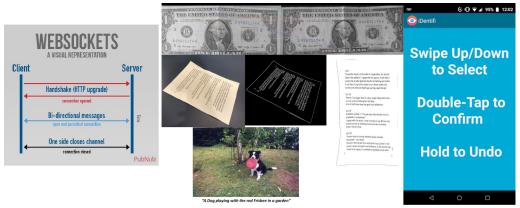


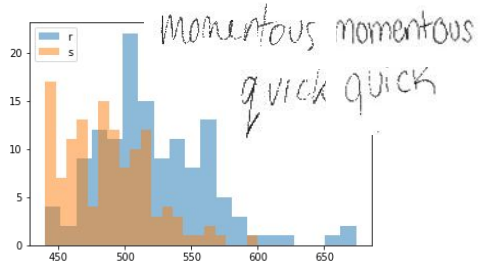
2021 - Analyzed student and school participation data from the past 8 years of MSEF (MA Science Fair). Used statistical tests and a custom semantic clustering algorithm, developing an R-Shiny application to allow them to repeat the analysis in future years.



2019 - Motivated by seeing crossing guards in my town braving weather and traffic, I created a prototype device that can be placed in a crosswalk and work as a crossing guard. Used YOLOv2 CNN to detect cars and people from 4 webcams, connecting my laptop to an Arduino UNO via PySerial to control the lights via relays.

2019 - My friend and I wanted to create an app for the visually impaired including features such as OCR, Scene Description, Money and Color ID. We got those working on our PCs via various deep learning frameworks, but struggled to implement them in an Mobile app. Although we didn't get all the features, we presented our prototype at an entrepreneurship camp.





2019 - Got free Azure compute, and was curious if an ML model could distinguish between different people's handwriting. I made a CNN that stacked two images of the same word, and determined if they were written by the same person. I collected data from my family members, using normalization and preprocessing techniques to beat human-level accuracy.

Thank you

For additional details please visit:

- Website: https://rohanbandaru.github.io/site/
- Github: https://github.com/rohan2017

My journey in electronics, robotics, and programming would not be possible if not for the support and mentorship of many.
Special thanks to:

- Asst. Prof. Chen Wang (Mentor, Carnegie Mellon AirLab)
- Dr. Glenn Allen (Aerospace and EE Club Supervisor, LHS)
- Prof. Rod Grupen (Mentor, UMass LPR)
- Dr. Ani Nguyen (Mentor, Verseau Therapeutics)
- Mrs. Rebecca Stendhal (Supervisor, MSEF)
- Mr. David Gutierrez (FLL Mentor, Empow Studios 2011-2015)
- Dr. Pratip Banerji (Advisor: CodingSafari.org)
- Ms. Tracy (Math Teacher, LHS)
- Ms. Finkelstein (8th grade Math Teacher)
- Team Irrational, LHS Aerospace Club
- My family and friends