Project ID: 331

SR - Engineering: Electrical, Mechanical, and Robotics

Angelina Kim
Grade 10
The Bishop's School
Advisor: Marcus Jaiclin

Deep Reinforcement Learning of PID Control for Rotational Wing Aerial Vehicles

A Deep Reinforcement Learning (DRL) in a mission controller has the potential to improve the performance of a Rotational Wing Aerial Vehicle (RWAV) with static Proportional, Integral, and Derivative (PID) controllers by updating the PID setup dynamically. The procedure is to implement a DRL network for Software-In-The-Loop (SITL) simulation of RWAV. The DRL network is trained by flight telemetry data including a gyroscope at 10Hz, and it dynamically updates PID controllers with their coefficients to hold a target position at 10m above the coordinate origin. The static PID RWAV's position-holding response under random wind direction and velocity is examined. With 5m/s wind, the DRL-engaged RWAV's training and evaluation processes are collected and compared. Two DRL engagement cases are experimented: 1) Rate PID controller only and 2) all position, velocity, attitude, and rate PID controllers. 5-layer DRL network with REINFORCE TensorFlow agent was used. In conclusion, DRL network with all-dynamic PID controllers' 19 coefficient updates improved position consistency by 48% at 0.26m from a static PID controllerâ€™s 0.5m under 5m/s random wind. The rate PID controller update with 8 coefficients degraded position consistency by 8% at 0.53m.

Project ID: 332

SR - Engineering: Electrical, Mechanical, and Robotics

Ronit Munshi
Grade 10
Canyon Crest Academy
Advisor: Devashish Munshi

Self Balancing Robot

An attempt to simulate a Segway, specially the way it balances on two wheels. In essence it is an inverted pendulum problem, with the center of mass above the wheels. The wheels must move to keep the pendulum balanced.

Engineering design
The basic design is two wheels being controlled by motors, with a vertical board above the wheels simulating the person on the scooter, which is a novel configuration. The Arduino board and battery are mounted on the vertical board. The motors are controlled by a balance shield that contains both the motor driver as well as the inertial unit consisting of an accelerometer and gyroscope.

Balance theory
If the robot gets tilted, its CG is no longer above its wheels and it will start to fall. The wheels must be spun forward / back to bring it back to its equilibrium position. To make this work, the wheel RPM is made proportional to the tilt angle.

Testing
Had to develop an awareness of the robot’s orientation and ensure that commands to spin the wheels worked correctly. Since the robot’s center of gravity is not exactly above the wheels, one had to determine the correct level of tilt to keep it balanced. The control algorithm has a couple of constants that needed to be found iteratively till a proper balance was obtained.

Design success
Currently the robot is able to balance and slowly move around its position. More work is needed for it to move forward and back.

Project ID: 333

SR - Engineering: Electrical, Mechanical, and Robotics

Nicholas Usher
Logan Brown
Grade 11, Grade 10
High Tech High North County
Advisor: Caroline Sailor

A Differentially-Geared, Omnidirectional Robotics Platform With Autonomous Jerk-Limited Navigation

Problem: Wheeled robotics research and commercial application development is inhibited by the lack of a fast, powerful, and easy-to-use robotics platform that can operate omnidirectionally in constrained spaces.

Solution: A differentially-geared robotics platform that combines the strengths of three traditional wheeled chassis (tank, mecanum/omni, and swerve) while eliminating their weaknesses. This platform includes custom software with autonomous 3rd derivative, jerk-limited navigation. It is built using off-the-shelf and easy-to-manufacture parts, increasing usefulness and attainability.

Process: Our CAD-centric, iterative mechanical engineering design process and hardware/software integration testing initially uncovered challenges preventing us from meeting our performance criteria. In addressing these problems, our solutions included:
1. Implementing an NVIDIA Jetson and Arduino based control system
2. Creating an asynchronous, binary, CRC error-checked communication system to enable loop speeds of less than 4ms
3. Redesigning the gear train to increase the robot velocity to meet the requirements (> 1 m/s)
4. Creating a fast and accurate navigation system that uses jerk-limiting to address the non-instantaneous nature of swerve drives

Conclusion: Our differential swerve drive platform is omnidirectional, has more power on average than mecanum/omni drives, and is 50% lighter than traditional swerves with the same power delivery. Its easy-to-use, versatile software enables both human and autonomous control. It allows the robot to smoothly navigate along paths to a goal position within an accuracy of 1 cm while traveling at its top speed of 1.4 m/s.

Project ID: 334

SR - Engineering: Electrical, Mechanical, and Robotics

Yash Patil
Grade 11
Sage Creek High School
Advisor: Sudhir Patil

Drunk Driving Recognition

This project aims to decrease the number of fatalities caused by drunk driving, which has been increasing and is responsible for around 10,000 deaths a year. The hypothesis is that a device powered by USB connection, which records accelerometer data such as speed and forces in different axes, will be able to detect if a driver is operating the vehicle in a dangerous manner. The data will be compared to a predetermined safe range, and if it exceeds this range, the device will use GPS data to inform emergency services about the dangerous driver. The objective of the project was to create a device that could detect drunk driving using GPS and G-force sensors, record if excessive force was applied to the vehicle, and alert emergency services to take the danger off the road, reducing the risk to everyone on the road. The device was designed using C++, Arduino, an accelerometer, a GPS module, a 3D printer, and a printer. The device was tested for performance and accuracy under different conditions, and the results showed that it met the desired specifications. The code was written in C++ and uses several libraries such as "Wire.h," "TinyGPS++.h," and "SoftwareSerial.h." The code collects accelerometer and GPS data and sends a signal with the vital data to alert emergency services in case of any dangerous driving behavior.

Project ID: 335

SR - Engineering: Electrical, Mechanical, and Robotics

Nitika Tatineni
Grade 11
Canyon Crest Academy
Advisor: Anthony Mauro

Energy-Saving HVAC System Using Ultrasonic Sensors and Machine Learning

Using data from ultrasonic sensors, and analysis of human presence in a particular room or building of an office or home, energy overconsumption in building heating, ventilation, and air conditioning (HVAC) systems can be reduced. The current design is able to identify rooms in a certain building with humans and rooms without humans using machine learning with data from ultrasonic sensors. The prototype simulates an optimized energy consumption for the HVAC systems in the building (air-conditioning or heating) by reducing usage in unoccupied rooms. The major physical equipment used includes four HC-SR04 ultrasonic detectors, a circuit/breadboard, corresponding wiring, a computer for performing the machine learning analysis, and an Arduino to control the circuit using the algorithm. The dataset was obtained through various placements of the simulated humans in the model room to indicate a person sitting down, standing up, lying in bed, etc. Results from three different machine learning methods (Support Vector Classifier, Random Forest Classifier, and Deep Learning Neural Network) were compared to each other, with the dataset being 90% split into training data and 10% split into validation data. The number of epochs was altered in order to improve training accuracy for the deep learning approach. The final design can either use the Deep Learning model with 200 epochs or the Random Forest Classifier and leads to at least 75% accuracy in the respective approaches. This was the initial requirement of the design, and therefore the model was successful in detecting the presence of humans for HVAC optimization. The trained model was used to run inferences using an Arduino Uno as a proof of concept.