Akshay Raman

akshay.raman01@gmail.com | 678-477-1024 | github.com/akshayra97 | akshayra97.github.io
Address: 5622 Darlington Rd, Pittsburgh, PA - 15217 | InActive DoD Secret

Education

Carnegie Mellon University Expected graduation date: May 2024
M.S. Research in Mechanical Engineering Robotics ***GPA: 4.0/4.0***
Relevant Courses: Machine Learning and AI (24-787), Planning For Robots (16-782), Optimal Control and RL (16-785), Computer Vision (16-720), Estimation and Detection (18-752)

University of Illinois at Urbana Champaign Graduated: Dec. 2019
B.S. in Aerospace Engineering with Minor in Computer Engineering ***GPA: 3.4/4.0***
Relevant Courses: UAV Navigation Control (AE 483), Introduction to Robotics (ECE 470), Digital Systems Lab (ECE 385)

Skills

Languages:
Python, C++, MATLAB, C, CMake, Bash, C#, Java, JavaScript, HTML, CSS

Technologies & Tools:
Git, GitLab, Linux, Docker, OpenCV, Pytorch, NX-10, SolidWorks, Latex, Optitrack, Microsoft Office, CI/CD

Experience

Sandia National Laboratories | *Software R&D Intern* May 2023 - Present

Segmentation Based Airplane Contrail Detection *Python, Computer Vision, GitLab, Bash*

- Produced a python pipeline to process GOES (GeoStationary Environmental Satellite) image data; extracted per frame pixel intensity data from netcd4 files to predict contrails in real time.
- Convolved multiple kernels over input images to create a filter response bank; Implemented k-means clustering to generate a segmented image dictionary and utilized minimum Euclidean distance to create a 'wordmap' of clusters on a predicting image.
- Implemented a Hough transform feature extraction algorithm in conjunction with a gradient detection algorithm to detect and classify 'thin' straight line clusters as contrails.

C++ HDF5 Satellite Frame Data Converter *C++, CMake, Docker, GitLab, Bash*

- Developed a HDF5 C++ converter to read, convert and write HDF5 satellite image data from WFOV (Wide Field of View) to FRM product. Increased conversion speeds for real-time processing by leveraging Sandia's established satellite data product libraries.
- Created test cases to validate structure and correctness of product conversion by adhering to CI/CD best practices.

Mechanical and AI Lab Research | *Prof. Amir Farimani - Carnegie Mellon* Sept. 2022 - Present

Autonomous Multi Agent Task Planning Charging Stations *Python, Computer Vision, Git, Bash, SolidWorks*

- Goal: Create a multi agent crazyflie wireless charging ecosystem to allow large scale aerial projects with limited battery charging stations. Leverage laser positioning systems with vision, AI, and planning algorithms to learn and guide multiple drones to respective charging stations while keeping atleast one drone in the air at all times.
- Designed an ecosystem of 'worker' drones interacting with 'bolt' (camera drones) drones to precisely locate open charging pads within 1cm accuracy. Configured lighthouse laser positioning system to ping current position coordinates to host(python) driver for planning (updating next state).
- Designed a custom frame with 2S motors and integrated a 'bolt' flight controller to carry heavier payloads (mini RGB camera).
- Deciphered the current location of an open charging pad within 5cm using a position calibrated camera along with 4 preplaced April Tags to locate a charging pad in the camera frame and convert into lighthouse coordinates.

Precognition of Low Battery and Position Error Mitigation During Low Battery *Python, AI, Git, Bash*

- Goal: Implement AI methods on crazyflie ouput parameter features to predict battery level ahead of time to allow drone to safely fly to charging pad.
- Collected a dataset of 35 features such as distance traveled, motor power used and battery statistics to predict current cycle battery dissipation.
- Interpreted collected features using T-sne and Random Forrest Regression models to analyze and classify most important features in reducing drone position error.
- Implemented an ANN regression model an aggregated loss function to predict the position error when travelling from position 'x' to 'y' on various battery levels. Trained model to reduce arrival errors from 12cm to <1cm when battery levels fall below 25%.

Boeing Aerospace April 2022 - Aug. 2022

Electrical and Navigation Systems Engineer *Python, C#, Bootstrap, Microsoft Office*

- Developed python scripts to test multiple Inertial Measurement Units and GPS Receivers to debug 'Could Not Duplicate' field failures and validate functionality.
- Conducted experiments to evaluate and enhance GPS receiver performance; Integrated C# codebase (software) with vector modulators (hardware) to simulate a jamming environment on raw live sky RF signals.
- Collected GPS receiver 'Carrier to Noise' data, Anti-Jam performance data, and kalman filter performance data over a GPS jamming plane to analyze GCU (Guidance Control Unit) performance.
- Supported flight test cards: monitored navigation data including satellite acquisition, tracking, and Carrier to Noise Ratio data to enhance GCU performance.
- Designed a test method to solve a power dissipation issue with GPS receivers during production testing.

Projects

Multi-Robot Motion Planning for Quadrupeds Nov. 2023 - Dec. 2023

Course Project - Carnegie Mellon *C++, ROS, CMake, Bash, Linux, Git, Motion Planning*

- Developed and implemented three innovative algorithms for quadruped robot motion planning: Sequential RRT-Connect, Joint State-Space RRT-Connect Planner, and Conflict-Based Search.
- Conducted extensive simulations using Quad-SDK, analyzing algorithm effectiveness in generating kino-dynamically feasible and collision-free trajectories in diverse terrains. Integrated changes into a existing ROS-based quadruped framework, contributing to an open-source project with substantial code development.
- Conducted an ablation study revealing that the Conflict-Based Search algorithm outperformed the other two methods, achieving a 17.5% shorter average path length and a 62.8% faster planning time compared to the Sequential RRT-Connect, and a 10.3% shorter path length with a 76.2% faster planning time compared to the Joint State-Space RRT-Connect Planner.

Live Motion Direction Estimation April. 2023

Course Project - Carnegie Mellon *Python, Computer Vision, AI*

- Spearheaded the development of a real-time camera motion direction estimation system, employing advanced image processing and machine learning. Compiled a diverse dataset of 4000 images across multiple platforms to ensure robust model training and avoid overfitting.
- Utilized innovative feature extraction techniques, including pixel change analysis and disparity map generation using CV2.stereoSGBM_create, to categorize motion into five categories: forward, backward, turning left, turning right, and stationary.
- Conducted comprehensive evaluations of various models, achieving a 74.7% accuracy with Naive Bayes Classifier, 93% with Support Vector Classification, 83% with Linear Neural Networks, and an impressive 94.6% accuracy with Convolutional Neural Networks (CNN), demonstrating the CNN's superiority in image-based motion direction analysis.

License Plate Detection Ablation Study Nov. 2022 - Dec. 2022

Course Project - Carnegie Mellon *Python, Computer Vision, AI*

- Trained a YOLOv5 network to extract the license plate's bounding box from an image of a vehicle.
- Implemented image contouring techniques with k-means unsupervised clustering algorithm to separate characters in a license plate image into individual character images.
- Analyzed different ML algorithms to find best character predicting accuracy: KNN-19%, CNN-67%, FNN-88%

Forest Fire Burn Size/Direction Prediction Model Sep. 2021

Course Project - UCLA Extension *Python, AI*

- Integrated a Random Forest feature selection model with a Bayesian regression algorithm (GridSearch tuned), reaching a burn size prediction accuracy of about 60 percent.
- Performed an ablation study on various FNN models to determine the optimal burn direction prediction model.

UAV Control Systems Project Oct. 2019 - Dec. 2019

Course Project - Univ. of Illinois Urbana-Champaign *C, MATLAB, Control Systems*

- Implemented C code, on Visual Studio, to parse sensor data from a quad-copter to estimate its current state.
- Integrated IMU force measurements and implemented a hard weighting between motion capture data and IMU sensor data to reduce overall sensor noise and bias leading to improved stability of quad-roter.