Renu Rajamagesh

Software Engineer | UBC Engineering Physics

+1 (503) 857-3918 | renu.rajamagesh@gmail.com | Renu-R | renu-rajamagesh

Skills

Programming: Java, Python, Rust, C/C++, MATLAB, Julia, SQL, Go, Bash, 8051 assembly, LaTeX
Frameworks: Git, Maturin, pyo3, Django, AWS EC2, ROS, Qt, Linux, TeamCity, TensorFlow, Jupyter, Gradle, Chrome dev tools

Education

University of British Columbia — *Vancouver, BC*

BASc in Engineering Physics — *Sep 2019 - May 2024*

- Cumulative Average of 88.7% and 109 credits earned. Achieved Dean's List honor and Trek Excellence Scholarship in all sessions
- **Coursework**: Machine Learning Project, Digital Signal and Image Processing, Applied Linear Algebra, Computational Optimization

Experience

Tesla Motors — *Palo Alto*

Software Engineering Intern, Chassis Controls — *May 2023 - Aug 2023*

- Contributed to a python steering validation pipeline to set up, run tests and log results using a HIL tester comprising of over 150 pytest fixtures.
- Led the integration of position sensors and thermal chambers with real time diagnostics using protocols such as CAN, TCP and RS-422 serial.
- Developed a mixed Python and Rust module that concurrently transmits and receives CAN data from multiple ECUs at over 500 kbps.

UBC Solar — *Vancouver*

Simulation Lead, Electrical team member — *Sept. 2019 - Ongoing*

- Led a team of 5 members to design a custom race simulation app for a road-safe solar powered race-car that secured 9th place at FSGP 2022.
- Created accurate software models for all components of the car by applying regressions to data from data sheets and component tests.
- Designed Race Strategy web app using Django and React and deployed using AWS EC2 server using real time data by polling a database.
- Designed PCBs such as GPS, telemetry, control and steering wheel boards. with technologies such as MCUs, CAN bus, radio and cellular modules.

Kardium Inc. — *Vancouver*

Cardiac Mapping Software Engineer — *May 2022 - Dec. 2022*

- Performed system level analytics using MATLAB and engineered data processing pipelines reducing procedure log extration times by 30%.
- Led the development of a contact sensing tool, designing a PCB and creating a desktop app that processes signals and displays real-time maps.
- Developed algorithms and filters to process data collected through sensors and visualizations and projections to map and navigate the left atrium.
- Prototyped algorithms for cardiac mapping using techniques such as PCA, noise handling, regressions and pointset registration in the R&D process.

UBC Centre for Teaching, Learning and Technology — *Vancouver*

Software Automation Coop — *Jan. 2021 - Apr. 2021*

- Developed scripts and applications, used widely by 4 faculties at UBC, to facilitate online instruction and learning during lockdown.
- Automated processes and integrated tools on learning management platforms using REST APIs, web drivers (Selenium), task schedulers etc.

Projects

Surface Meshing for Medical Scans — *Sep 2022 - May 2023*

- Contributed to medical physics software *Dicomatomaton* by creating a C++ project that generates 3D surface meshes given planar contours.
- Implemented graph based tiling to find the optimal triangulation of points and used Dijkstra's algorithm for graph searching.
- Reduces run-time by 30% by implementing optimal computational geometry algorithms for convex hull tracing and bounded volume computations.
- Final algorithm produced meshes from CT scans with 10x fewer vertices compared to the marching cubes method for the same mesh accuracy.

Noteation: Eyetracking for Music — *Sept 2022*

- Hack the North Finalist project (placed top 12 in 200+ teams) that allows users to annotate sheet music and flip pages with eye gestures.
- Used AdHawk's eye-tracking glasses to stream events to CockroachDB using a python backend and the React frontend polls for cues to flip pages.
- Experimented with Polynomial Regression using scikit-learn to characterize gaze data collected.

Self-driving Car in a Simulated Environment — *Sep-Dec 2021*

- Developed the AI for a self driving car using ROS which navigates a *Gazebo* simulated environment to detect license plates and avoid collisions.
- Used OpenCV to detect the street, pedestrians etc. experimented with reinforcement learning and control algorithms to navigate the arena.
- Built a CNN architecture using tensorflow to recognize the characters on the license plates and wrote scripts to automate the process of data generation to train the neural network collecting over 10,000 images and achieving an accuracy of 99%.

Autonomous Can Collecting Robot — *May-Aug 2021*

- Designed and built an autonomous robot from scratch, that can locate and retrieve cans, reorient them and drop them into allocated slots.
- Wrote firmware in C for an STM32 MCU to control DC and servo motors based on inputs from IR sensors. Implemented a state machine for the robot.
- Optimized the driving performance, interpreting sensor data and fine-tuning PID control parameters to follow 15m of tape in under 30s.