Mark Do

+1 647-766-6794 | mdo@uwaterloo.com | github.com/hepromark | linkedin.com/in/markdouw/

Education

University of Waterloo - *Candidate for* ***Bachelor of Mechatronics Engineering*** 2022 - 2027

- 4.0 Cumulative GPA | 3 x Dean's Honours List

Work Experience

Algorithms Software Developer| *Python, SimPy* Sep 2024 – Dec 2024
Rapyuta Robotics Tokyo, Japan

- Created an end-to-end pipeline that generates optimized Robotic Storage System layouts based on customer requirements.
- Designed and implemented a custom event-based simulation platform with SimPy using spatial-temporal grids, achieving 12× speedup and lower memory usage; deployed to evaluate 5 new system improvements.
- Utilized Genetic Algorithms for non-convex optimization, resulting in 10% higher storage density without performance drops.

Robotics Software Developer| *Python, Django, SQL, ROS* May 2024 – Sep 2024
Rapyuta Robotics Tokyo, Japan

- Developed a Task Planner component to reactively schedule tasks for robots in a Multi-Agent Robotic Storage System, which translates customer warehouse operations into back-end task trees appropriate for parallelized robot execution.
- Scaled Task Planner & world model database to work with 10 times more robots (6 → 60) by using caches & lighter queries.
- Created 5+ robot error recovery behaviours to reduce MTBF from hours to days, enabling the system's initial rollouts.
- Oncalled for startup's first field deployment at a live warehouse, debugged 20+ critical issues across the stack in real-time.

Test Automation Developer | *Java, Docker, TestNG* Sep 2023 – Dec 2023
Definity Financial Waterloo, Canada

- Overhauled Java test suite with > 120 test cases for company's mobile app, reducing false positive bug detections by > 95%.
- Created a new automated CI pipeline for regression tests, executing & reporting results daily to save >12 dev hrs./week.

Related Experience

Computer Vision Researcher | 3D Reconstruction Apr 2025 - Present

- Researching novel Structure-from-Motion models with Vision Transformer (ViT) backbones for visual SLAM & 3D reconstruction of construction sites using drone-based imagery at Waterloo's Vision and Image Processing Lab.

Autonomous Racing Team Lead | *C++, Docker, ROS2, Linux* Jan 2025 - Present
F1Tenth Autonomous Racing Team

- Leading a team of 15+ undergraduates to develop a self-driving software stack for F1Tenth Autonomous Racing Competition.
- Led the design & implementation of pure pursuit control, lattice motion planner, offline & online SLAM algorithms.
- Built a dockerized dev environment integrating ROS2, Foxglove and AutoDRIVE physics simulator for testing algorithms.

Autonomous Vehicles Developer | *C++, Casadi, Kubernetes, Terraform, PyTree* Nov 2023 – Jul 2024
WATONOMOUS (Waterloo Autonomous Vehicles Design Team)

- Designed a dual-model MPC system, joining a Multi-Layer Perceptron with a traditional kinematics model for car navigation.
- Designed a high-level car navigation controller that uses behaviour trees to take driving actions based on GPS data.
- Maintained SLURM & SSH server infrastructure for over 400 developers across 3 universities for ML research & development.
- Created an Asset Manager that manages website assets using self-hosted Ceph S3 Buckets & deployed on a Kubernetes Cluster.

Technical Skills

Languages: Python, C, C++, Java, JavaScript, Bash, Arduino, MATLAB, Assembly

Libraries & Frameworks: ROS, Gazebo, Foxglove, RVIZ, PyTorch, OpenCV, Scikit-learn, PyTree, PostgreSQL

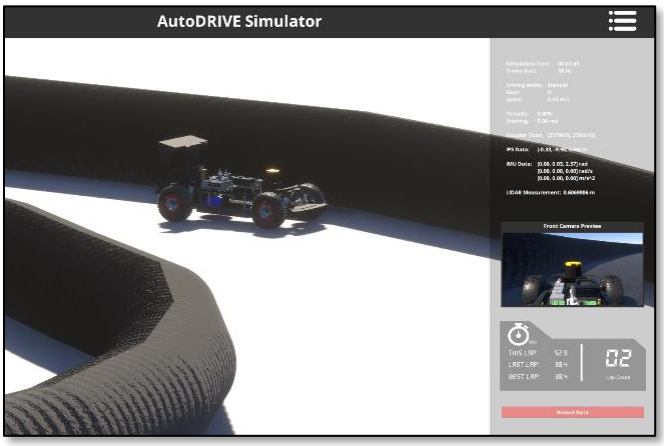
DevOps: Docker, Kubernetes, Terraform, Ansible, Git, GitHub, BitBucket, AWS

Notable Courses: Operating Systems, Data Structure & Algorithms, Advance Linear Algebra, Numerical Method

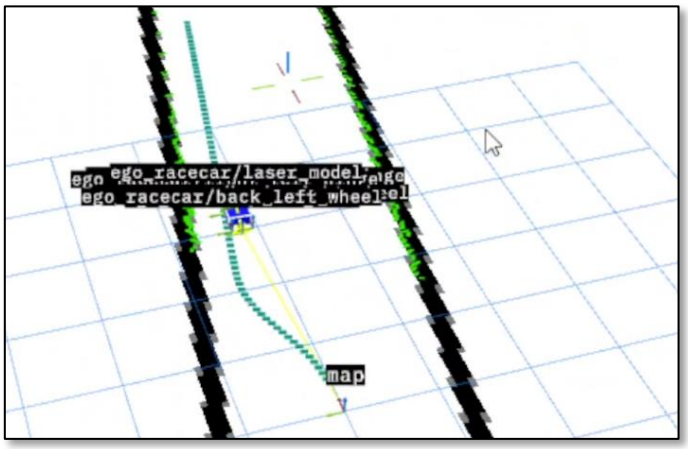
Mark Do - Portfolio

647-766-6794 | mdo@uwaterloo.com | github.com/hepromark | linkedin.com/in/markdouw/

F1Tenth Autonomous Racing Team Lead



F1Tenth Vehicle Simulator



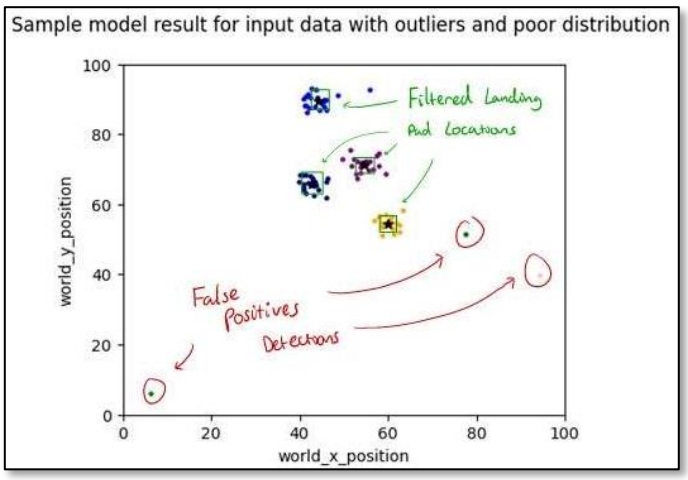
Pure-pursuit control applied to a spline generated by RRT

The F1Tenth competition is a 1/10th scale autonomous racing cup where global teams compete for the fastest autonomy algorithms.

- Leading a team of 15+ undergraduates to develop a self-driving software stack for F1Tenth Autonomous Racing Competition.
- Modelled the onboard software architecture after the See-Process-Act paradigm, prioritizing real-time control & safety.
- Led the design & implementation of pure pursuit control, lattice motion planner, offline & online SLAM algorithms.
- Built a dockerized dev environment integrating ROS2, Foxglove and AutoDRIVE physics simulator.
- Project Repo can be found here.

Landing Pad Clustering Algorithm: In-flight unsupervised learning model

WARG's 2023 Competition Drone: ICARUS

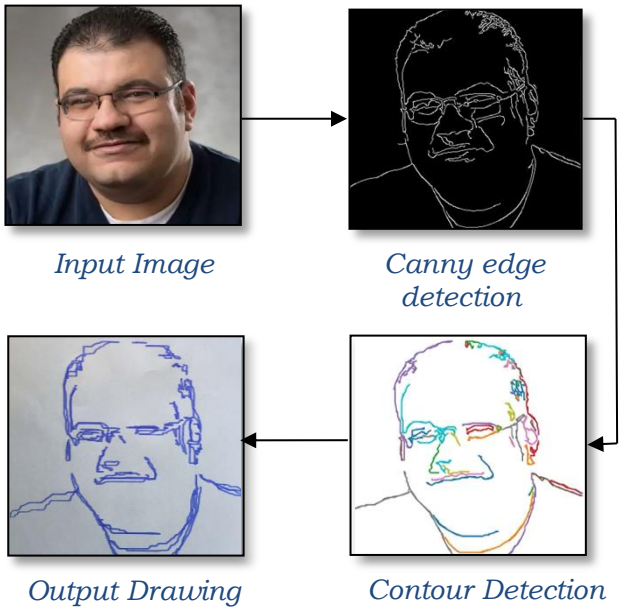


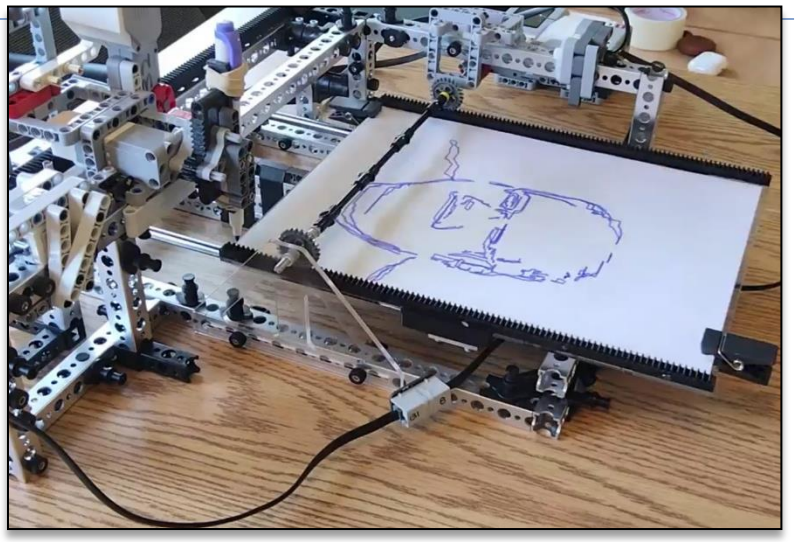
Clustering Algorithm Output: False positives are discarded from potential landing pads.

I worked on WARG's in-flight unsupervised learning model for the 2023 AEAC Competition

- Developed a clustering module for an autonomous drone to cluster landing pad detections & predict actual pad locations.
- Implemented a Variational Gaussian mixture model to consume pad locations over time and continuously output estimates.
- Created filters with dynamic thresholding to remove outlier pad locations, enabling drone to operate in poor visibility weather.
- Designed an intelligent memory system to bias towards new observations while retaining knowledge of pads only seen once.
- Code can be found here

Articus Maximus: Sketching Robot

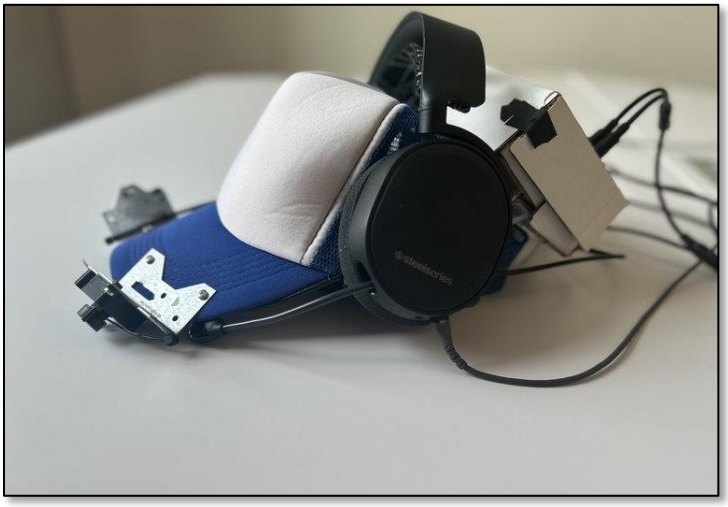




Sketching Robot after completing a drawing

- Designed & manufactured a 2-axis gantry sketching robot which controls a pen to draw images on paper from digital file input.
- Developed an image pipeline that extracts drawable contours from digital images with edge detection & contour detection.
- Optimized robot drawing speed by using the Douglas Peucker algorithm and Hu Moments to reduce total contour count.
- Programmed firmware PID controller in C with anti-windup & a 1D motion profile to draw lines accurate within 2 degrees.
- Code can be found here.

BlindWatchers: Assistive Vision Headwear

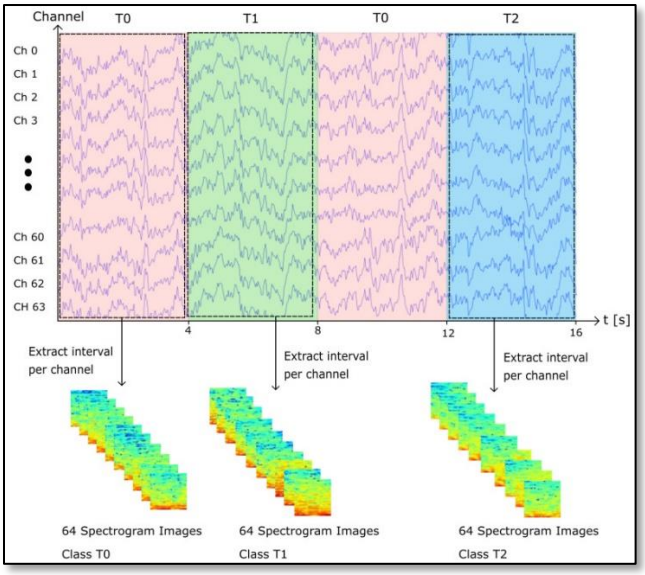


Headwear has 2 cameras, headphones for audio output, with an NVIDIA Jetson attached behind for compute.

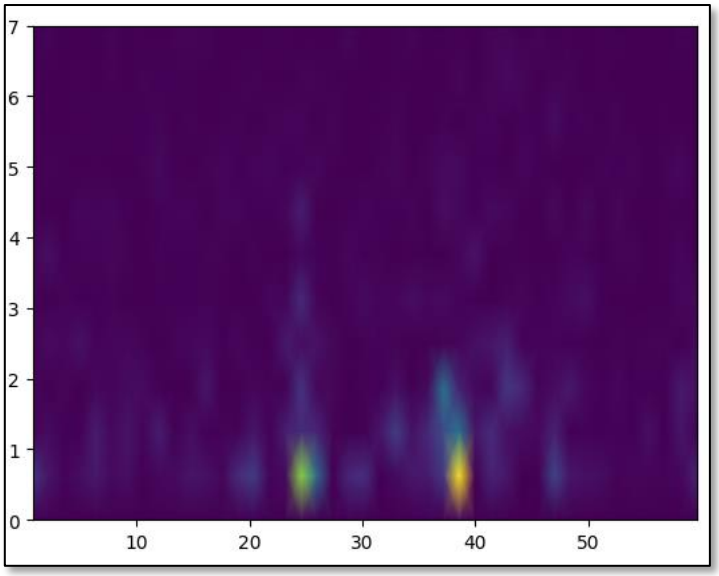
Sample YoloV8 detection bounding boxes, before conversion to spatial audio.

- Designed and developed a smart helmet for the visually impaired with an NVIDIA Jetson, integrating YOLOv8 object detection, directional audio, & Google Cloud Speech-to-Text to provide real-time auditory descriptions of nearby objects.
- Estimated object real-world 3D poses from 2D bounding boxes to provide matching directional audio for more intuitive UX.
- Implemented an asynchronous architecture with a fast-inferencing computer vision model to decrease latency between detection & audio.
- Code can be found here

EEG Machine Learning Model: Motor Imagery Classifier



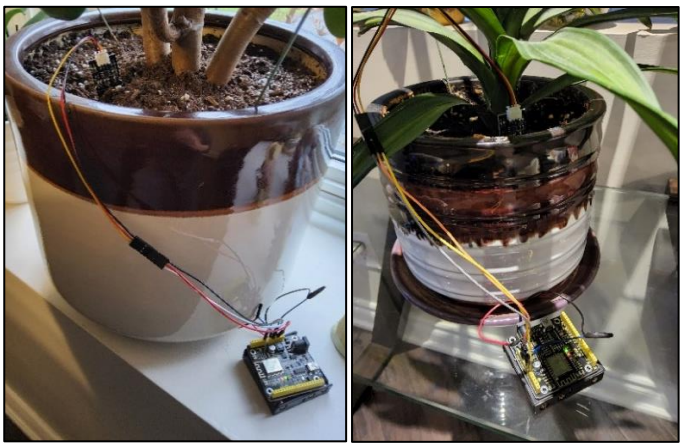
CNN Architecture takes 3D input volumes, created from stacking 64 electrodes' spectrogram images



Spectrogram image produced from a single electrode's 2-minute EEG segment

- Developing deep learning model on EEG brain data for motor imagery classification task on a 64-channel Motor Imagery dataset.
- Converted 64-channel time series data into volumes in the time-frequency domain using FFTs for use with convolutional kernels.
- Wrote custom data samplers in PyTorch with weak shuffling to optimize data read speeds for training batches from .h5 files.
- Project Repo can be found here

Soil Humidity IoT: Cloud-based agriculture monitoring.



Examples of soil-humidity detector ESP8266 setup

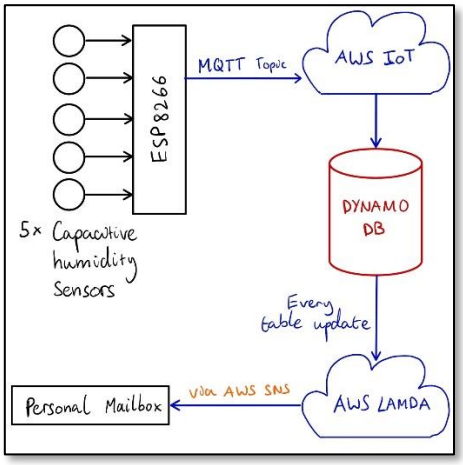


Diagram of the agriculture monitoring system.

- Designed and created a monitoring system to track houseplant soil humidity, alerting users via SMS when below set threshold
- Programmed ESP8266s to publish humidity levels via Wi-Fi to an AWS IoT MQTT topic, storing the data in Dynamo DB.
- Created an AWS Lambda function listening to database writes, to alert users using AWS SNS if readings below threshold.
- Code can be found here