Mark Do

647-394-6627 | mdo@uwaterloo.com | github.com/hepromark | linkedin.com/in/markdouw/

Education

University of Waterloo Sep. 2022 – May 2027

Candidate for Bachelor of Mechatronics Engineering

- 4.0 Cumulative GPA. Dean's Honors List: 2022 & 2023

Technical Skills

Languages: ROS2, Python, C/C++, Scala, Java, Git, SQL, Arduino, MATLAB, Dart

Libraries & Frameworks: Docker, TensorFlow, PyTorch, OpenCV, Scikit-learn, TestNG, Appium

Design & Prototyping: Siemens NX, SolidWorks, AutoCAD, 3D-Printing, Laser-cutting, Soldering, Oscilloscope

Work Experience

QE Automation Analyst | *Docker, CI/CD Pipelines, Java, TestNG, Appium, Gradle* Sep. 2023 – Dec. 2023

Definity Financial

- Enhanced regression suite with over 120 test cases for Sonnet's app on iOS & Android, increasing test suite reliability.
- Developed data-driven automation modules for test suite using TestNG via requests to BrowserStack's REST API endpoints.
- Created a new CI pipeline by Dockerizing automated test suite & Allure reporter for deployment to Bitbucket pipelines.
- Leveraged automatic pipeline triggers to eliminate need for manual test startup or reporting, saving > 12 hours weekly.

Production Technician | *Data Analysis, Scikit Learn, Pandas, NumPy, System Optimization* Jan. 2023 – Apr. 2023

Olymel S.E.C

- Performed dimensionality reduction, regression & analysis of production data in Python to optimize manufacturing process.
- Created a data transformation pipeline in Python to process raw .CSV production data into smaller Pandas Data-frames.
- Aggregated and analyzed 2 years of production data using pipeline, leading to 3 recommendation reports from analysis.

Open-source Experience

Autonomy Developer | *Open-source, Scikit-learn, Linux, NVIDIA Jetson, Multi-processing* Apr. 2022 – Present

***WARG** is a design team focused on the unmanned aerial vehicle industry, which develops open-source autonomous flight software.*

- Developed a clustering algorithm for WARG's perception-decision-controls system to predict drone landing locations.
- Utilized a Variational Gaussian mixture model to group detected landing pad locations and find the centers of data clusters.
- Integrated the module into producer-consumer multi-processing model, ensuring correct functionality through unit tests.
- Performed hardware-in-the-loop testing for landing pad detection module with NVIDIA Jetson on a flying drone.
- Created post-mortem test reports after drone flights and usage documentation for cluster module.

Projects

Articus Maximus - Sketching Robot | *C, PID Controller, Image Processing, OpenCV* GitHub

- Designed & manufactured a 2-axis gantry sketching robot which controls a pen to draw images on paper from digital file input.
- Created an image processing pipeline in Python to pre-process, detect edges, and find contours from .JPG or .PNG file types.
- Implemented translational-variant Hu Moments and a recursive Douglas-Peucker algorithm to remove redundant contours.
- Programmed and tuned a PID controller in C with anti-windup and a 1D motion profile to draw lines accurate within 2 degrees.

Emotional Cardiography (ECG) | *OpenCV, TensorFlow, Flask, Python, C++, JavaScript* GitHub

- Developed and trained Convolutional Neural Network using Keras-TensorFlow, achieving an accuracy of over 70% in recognizing 5 emotions. Integrated with Flask, the model transfers the real-time emotion predictions to a NodeJS backend.
- Iteratively improved model performance using error analysis, image augmentation, and skip connections inspired by ResNet.
- Designed and programmed Arduino C/C++ model to detect users' heartbeat, using serial communications to transfer data.