

# Joseph Mok

hojinmok@gmail.com • (647) 975-5126

Portfolio Website: [https://hjmok.github.io/josephmok\\_portfolio](https://hjmok.github.io/josephmok_portfolio)

Linked In: <https://www.linkedin.com/in/hojin-joseph-mok-31153a163/>

GitHub: <https://github.com/hjmok>

---

## Summary of Qualifications

- Proficiency in utilizing TensorFlow/Keras, PyTorch, Scikit-Learn, OpenCV, NumPy, and Pandas libraries in Python
  - Programmed various machine learning models including linear/logistic regression, CNN, RNN, NLP models, etc.
  - Well versed in data preprocessing and feature engineering in preparation of machine learning model training
  - Experienced with SCADA and PLC software packages, including Allen Bradley and Ignition Automation Software
  - Experienced integrating MySQL, Python scripting, and Ignition for capturing SCADA trend data and data analysis
  - Strong cross-functional project management experience from leading various multidisciplinary projects
- 

## Professional Experience

Yaya Foods Corp.

Toronto, ON

**Senior Automation Engineer**, July 2019 – Present

- Lead the automation design for several SCADA systems for beverage manufacturing clients by integrating Rockwell Automation PLCs/HMIs, Ignition Automation Software, MySQL, and ethernet IP networks
- Developed data mining process by storing transmitter data from PLCs to MySQL database using Ignition SCADA packages. Developed client applications within Ignition for data analysis (heavy Python scripting)
- Gained strong troubleshooting skills for hardware and software to determine root cause of production halts
- Sourced transmitters, motors, VFDs, valves, and necessary hardware for new automation systems

Apple Inc.

Cupertino, CA

**Recycling R&D Engineer Intern**, September 2018 – April 2019

- Managed development of recycling systems for Apple products. Involved heavy coordination with vendors and internal Product Design teams to organize documentation and deadlines
  - Developed optimal cycle time processes for de-manufacturing of various Apple products. Resulted in a 90% increase in the Unit-per-Hour output compared to the current manual methods
  - Designed semi-autonomous prototypes to showcase cycle time and disassembly improvements. Designs continuously integrated feedback to improve operator ergonomics and meet California waste regulations
- 

## Projects

The following are Computer Vision projects. To see all projects, please visit: [https://hjmok.github.io/josephmok\\_portfolio](https://hjmok.github.io/josephmok_portfolio)

### Generative Adversarial Networks for Dogs & Cats

[https://hjmok.github.io/josephmok\\_portfolio/#/GAN](https://hjmok.github.io/josephmok_portfolio/#/GAN)

- Used PyTorch to create a GAN that outputs images of fake dogs and cats
- The Discriminator was trained on 24994 prelabelled photos from Kaggle's Dogs vs. Cats Dataset
- Generator was able to product distorted images of dogs & cats, which can be seen in the results section

### Cat vs Dog Classifier

[https://hjmok.github.io/josephmok\\_portfolio/#/CNN](https://hjmok.github.io/josephmok_portfolio/#/CNN)

- Created Convolution Neural Network models on both Tensorflow and PyTorch to classify Dogs and Cats
- Used dropout layers and Batch normalization on PyTorch model which greatly reduced overfitting. Tensorflow model achieved a 92.6% training accuracy and 78.6% validation accuracy. However, PyTorch model achieved a 77.5% training accuracy and 76.6% validation accuracy.
- Implemented the AlexNet model, which improved results with 91% training accuracy and 95% test accuracy

### Object Detection with Single Shot MultiBox Detection

[https://hjmok.github.io/josephmok\\_portfolio/#/SSD](https://hjmok.github.io/josephmok_portfolio/#/SSD)

- Implemented a pre-trained SSD model using PyTorch to perform Object Detection on videos
- Created a Detect Function to take input video frames and convert them into Torch variables that the SSD model accept. Then used OpenCV to apply red rectangles around the detected objects. Video in link above

### Face Detection with Haar Cascades

[https://hjmok.github.io/josephmok\\_portfolio/#/FD](https://hjmok.github.io/josephmok_portfolio/#/FD)

- Used OpenCV to load pre-trained Haar Cascades for the face, eyes, and smile
- Created a Detect function which applies rectangles around the regions of interest detected by the Haar Cascade to the input video frames. Performed face recognition using the user's webcam. Video in link above

### License Plate Detection and Blurring

[https://hjmok.github.io/josephmok\\_portfolio/#/license\\_plate](https://hjmok.github.io/josephmok_portfolio/#/license_plate)

- Loaded a pre-trained Russian License plate Haar Cascade using OpenCV
- Created a Detect and Blur function which animated a rectangle around the region of interest detected by the Haar Cascade, then applied a median blur to said region

---

## Education

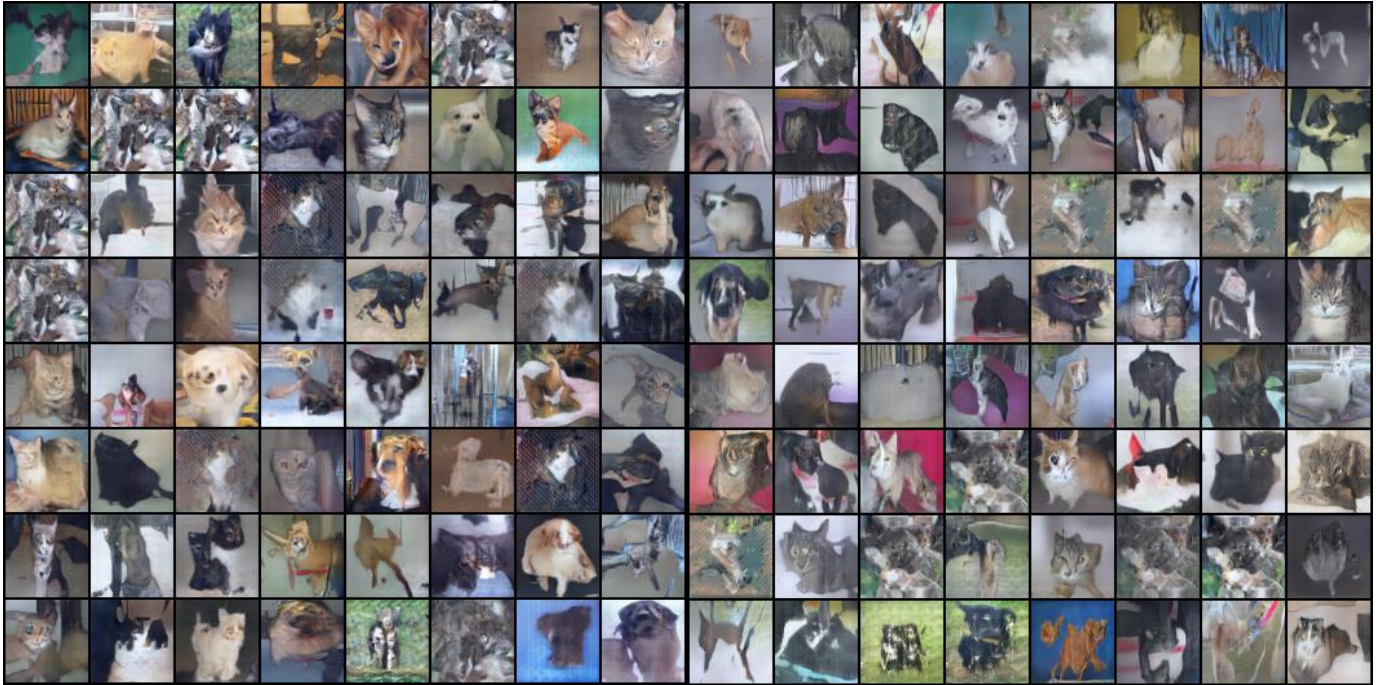
---

University of Waterloo,  
**Bachelor of Applied Science, Honours Mechanical Engineering**, Graduated June 2019

## Project Result Images

### Generative Adversarial Networks for Dogs & Cats

[https://hjmok.github.io/josephmok\\_portfolio/#/GAN](https://hjmok.github.io/josephmok_portfolio/#/GAN)



### License Plate Detection and Blurring

[https://hjmok.github.io/josephmok\\_portfolio/#/license\\_plate](https://hjmok.github.io/josephmok_portfolio/#/license_plate)

