Joseph Mok

Email: hojinmok@gmail.com GitHub: github.com/hjmok
Portfolio: hjmok.github.io/josephmok_portfolio LinkedIn: linkedin.com/in/hojinjosephmok

Skills

Languages:	Python, JavaScript, MATLAB, C++
ML Libraries:	TensorFlow/Keras, PyTorch, Scikit-Learn, NumPy, Pandas, spaCy, NLTK, OpenCV, PySpark
Databases:	MySQL, SQL, HBase, Cassandra
Big Data:	Hadoop, Pig, Hive, PySpark

Professional Experience

Yaya Foods Corp. *Toronto, ON*

Automation Engineer, July 2019 – Present

- Developed data mining process that extracts PLC transmitter data, stores in MySQL database, and presents it to Client application designed in Python and Ignition Automation software
- Lead the automation design for several SCADA systems for beverage manufacturing clients by integrating Rockwell Automation PLCs/HMIs, Ignition, MySQL, and ethernet IP networks
- Determined root cause of production halts through strong troubleshooting skills for hardware and software

Apple Inc. *Cupertino, CA*

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

- Designed semi-autonomous prototypes for optimal disassembly and recycling of Apple products. Designs continuously integrated feedback to improve operator ergonomics and meet California waste regulations
- Prototype resulted in a 90% increase in the Unit-per-Hour output compared to the current manual methods

Projects

Generative Adversarial Networks for Dogs & Cats *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

Convolutional Neural Network for Cat vs Dog Classifier *hjmok.github.io/josephmok_portfolio/#/CNN*

- Created CNN models on both Tensorflow and PyTorch to classify Dogs and Cats
- Tensorflow model achieved a 92.6% training and 78.6% validation accuracy. PyTorch model achieved a 77.5% training and 76.6% validation accuracy after adding dropout layers and batch normalization
- Implemented the AlexNet model, which improved results with 91% training accuracy and 95% test accuracy

Object Detection with Single Shot MultiBox Detection *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

License Plate Detection and Blurring *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

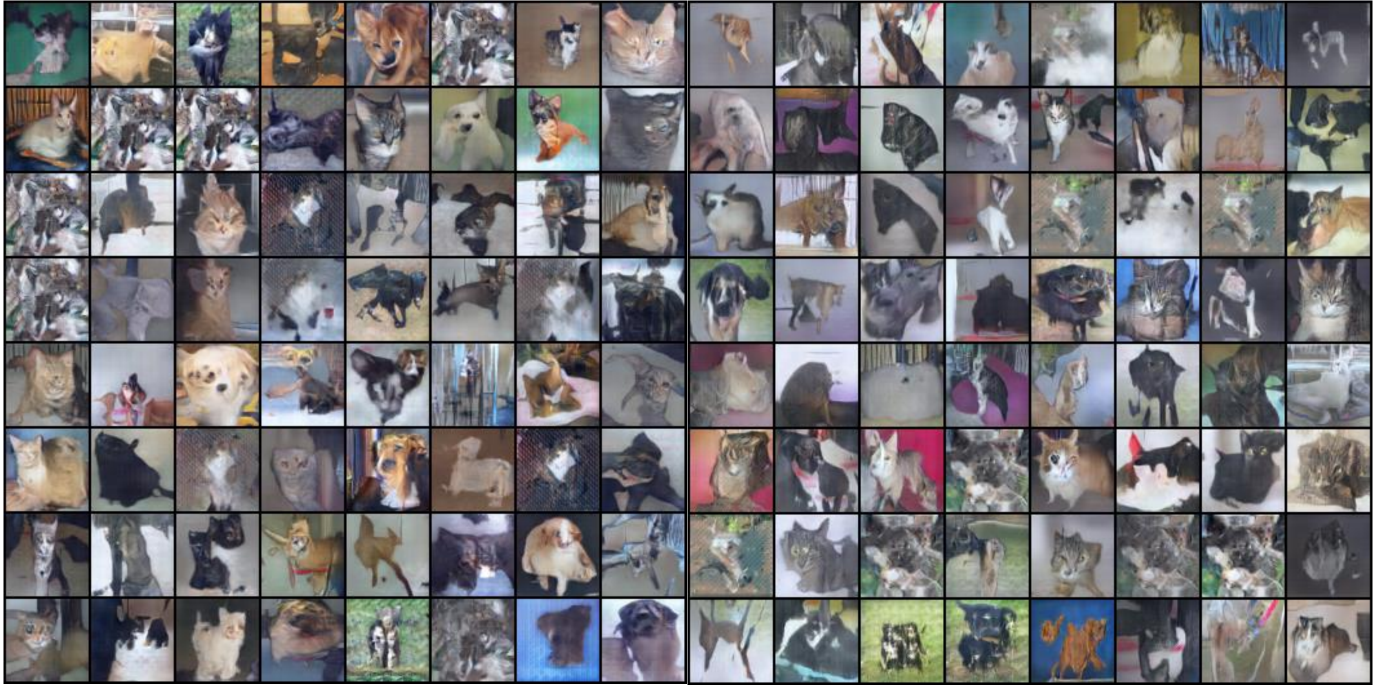
Education

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

Project Result Images

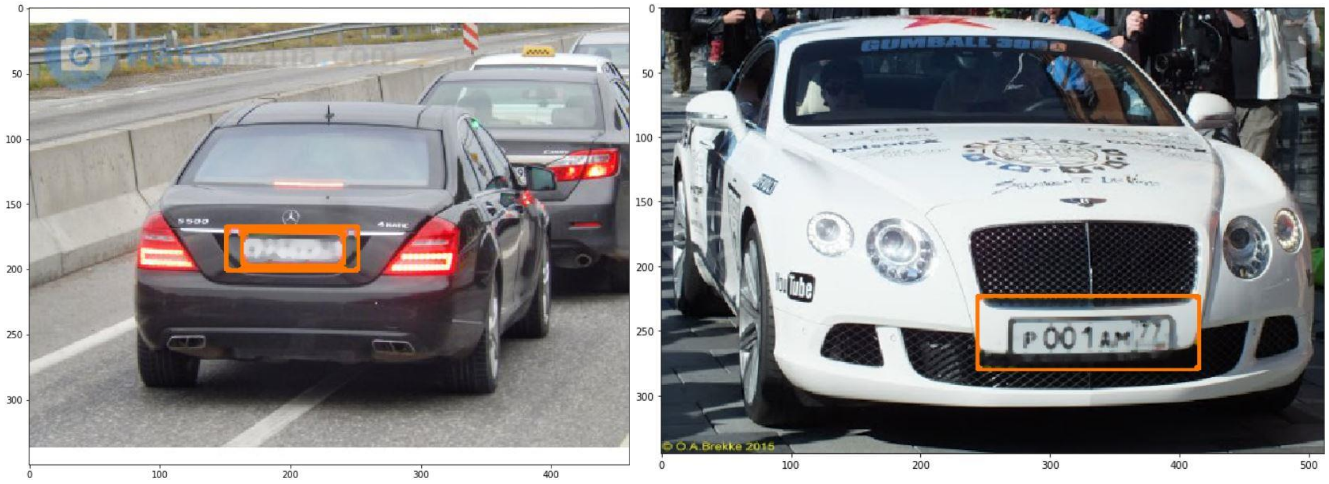
Generative Adversarial Networks for Dogs & Cats

hjmok.github.io/josephmok_portfolio/#/GAN



License Plate Detection and Blurring

hjmok.github.io/josephmok_portfolio/#/license_plate


GUMBALL 3000
P 001 AM 22
© O.A.Brekke 2015