

Jerome Siljan

JeromeSiljan@gmail.com • (469) 733-8585 • [GitHub](#) • [LinkedIn](#)

Software Engineer | Data Analyst

Core Competencies:

- Cleaning data
- Natural language processing with machine learning
- Computer vision with Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), and traditional techniques such as masks
- Cloud hosted databases
- Linear Regression
- Time Series Analysis

Technical Skills:

Languages: C, C++, Python, Java, ARM Assembly, Verilog (HDL), HTML, CSS

Applications: Arduino IDE, Quartus Prime, OnShape, Fusion360, Ultimaker Cura

Tools/Libraries: tensorflow, opencv, pandas, matplotlib, numpy, flask, CockroachDB, SQLite3

Certifications: IBM's Analyzing Data with Python (DA0101EN)

Education:

Bachelor's of Science in Computer Engineering at The University of Texas at Arlington

December 2023

Projects:

Monocle

- This project was built to solve CBRE's challenge from the [2022 Texas A&M Datathon](#).
- Used machine learning to identify, read, and cluster text in order to sort them into sequences using **CRAFT** and **tesseract**.
- Designed and implemented a custom clustering algorithm that groups text based on their coordinates.
- Increased OCR accuracy from 2% to 80% by including text detection (using **CRAFT**) in the preprocessing stage.
- Developed **flask** web UI that interfaces with the OCR component to display results.
- Successfully organized and managed the work load between a team of engineers in order to complete the project in a tight timeline.

Pistachio Detective

- Classified images of pistachios as being either Kirmizi or Siirt pistachios using a convolutional neural network in **tensorflow**.
- Accurately classified pistachios with an 85% accuracy with only 2148 images and a 9/1 training/validation split.
- The Pistachio Detective uses a convolutional neural network to both speed up compile times and minimize the impact of "noise" in the images. Dropouts were used to combat overfitting.
- Methodically used machine learning fundamentals, such as producing an appropriate training/testing set balance, applying different techniques to avoid overfitting, and methods more specific to image classification, such as CNNs and image augmentation to achieve a high accuracy.