Andy Zhang

(408) 839 8887
jz359@cornell.edu

1641 Deerfield Dr.
San Jose, CA 95129

EDUCATION

Cornell University 2016 – 2019
B.S., Computer Science // Minor, Electrical Engineering
Relevant Coursework: Honors Data Structures // Signal Processing // UNIX Scripting // Functional Programming
Concentrations: Data-Intensive Computing, Signal Processing

EXPERIENCE

Computer Vision Intern Summer 2017
SRI International
Responsibilities include developing and testing algorithms for Simultaneous Localization and Mapping (SLAM) using ROS and Python. Also wrote a program to detect, identify, and extract facial features of people in a given video. Used Docker to make software cross-platform.

Computer Vision Developer 2016 – present
Cornell Unmanned Air Systems (CUAir)
CUAir is a project team that designs, builds, and tests an autonomous aircraft system for the Student Unmanned Air Systems (SUAS) Competition. Currently working on detection, segmentation, and classification modules as part of the Computer Vision subteam.

Morphometrics Research Intern Summer 2015
University of California, Santa Cruz
Researched the trends in the morphology of nautiloids and ammonites using Fourier analysis and Principal Components Analysis. Conducted under the supervision of Prof. Matthew Clapham and mentor Dan Killam.

PROJECTS

modemo (in progress) Summer 2017
Web application using NLP to detect and quantify political bias in articles. Made with Python, React/JS, Keras, and Theano.

baeML Summer 2017
Web application using NLP to offer personalized content designed to counteract the echo-chamber effect of social media. Key components include a React frontend, Skip-gram learning model, database, and webcrawler. Made with Python, React/JS, and Tensorflow.

Cell_ID Spring 2017
Computer vision project using OpenCV and Python to process images of white blood cells and classify them as one of five types to detect and diagnose blood-related diseases.

Critter World Fall 2016
Simulation of a world with "critters" modeled by a custom language, compiler, interpreter, and GUI. The world is maintained by a server, and multiple clients connecting to the world can request updates to the world state, which is tracked by a diff. Made with Java.

SKILLS

Languages
Java (5/5), Python (4/5), Javascript (3/5), HTML/CSS (3/5), MATLAB (3/5), R/RStudio (2/5)

Software and Tools
Git, Linux, UNIX

Software Libraries
OpenCV, Scikit-learn, Tensorflow, Django, PostgreSQL, ROS