

David Tsai

dtsai@berkeley.edu
408.480.8634
github.com/tasilb • tasilb.me

Education	University of California, Berkeley <i>Bachelor of Arts, Computer Science (expected May 2017)</i>	Aug. 2013 - present
Coursework	Machine Learning, Artificial Intelligence, OS and Systems Programming, Databases, Algorithms, Data Structures, Computer Architecture, Microelectronic Circuits, Linear Algebra, Probability Theory, Engineering Statistics, Robotics (in progress), Embedded Systems (in progress), Computational Photography (in progress)	
Experience	Project A. R. T. Research Developer <i>Machine Learning@Berkeley</i> Developing vision and modelling software for a autonomous garment tailoring project. Implemented a contour detection script using OpenCV.	Mar. 2016 - present
	Core Leadership <i>Robotics@Berkeley</i> Providing technical, logistical advising for projects involved in the R@B sponsorship process. Designed and hosted technical workshops in topics related to robotics.	Sep. 2015 - present
	Software Engineering Intern <i>MosaixSoft, Inc. - Los Altos, San Francisco CA</i> Developed cloud infrastructure software for OpenStack using Java at a stealth startup. Used JAX-RS and OpenStack APIs to defragment an existing component, expand our product's capabilities and make future engineering around the component more transparent and flexible.	May 2016 - Aug. 2016
	Undergraduate Research Assistant <i>UC Berkeley</i> Built requisite software for projects and papers under Eric Paulos, worked on OnePhoto, an Android application that takes a single photo and only displays it thereafter.	Sep. 2015 - Feb. 2016
	Academic Intern - CS 61A <i>UC Berkeley</i> Assisted with teaching programming concepts and Python topics to students. Coached students through problem-solving techniques on assignments and debugged their projects and homework during office hours.	Jun. 2015 - Aug. 2015
Projects	MNIST SVM digit classifier (CS 189 project, Spring 2016) Entered an in-class Kaggle competition to classify digits from the MNIST handwritten digit dataset. Implemented a linear SVM classifier using scikit-learn. Employed 10-fold cross-validation for tuning the regularization hyperparameter.	
	PintOS (CS 162 project, Fall 2015) Enhanced the functionality of the PintOS dummy OS in C. Added priority-based scheduling and a multi-level feedback system to the thread scheduler. Implemented a Unix-like inode file system. Enabled the operating system to run user programs.	
	Sliding tile puzzle solver (CS 61C project, Fall 2014) Wrote a program to strongly solve sliding tile puzzles. Parallelized a breadth-first algorithm using MapReduce in Apache PySpark by partitioning the tree traversal of the problem space over multiple workers. Deployed to Amazon EC2 instances provided by the course staff and benchmarked it.	
Skills	Programming Languages: Python, C, Java, JavaScript, MIPS, x86 Web: Ruby on Rails, Meteor, jQuery, HTML/CSS, Jekyll, Flask, JAX-RS Tools & Platforms: Linux, Git, PostgreSQL, Neo4j, Apache Spark, Amazon Web Services, Docker, OpenStack, NumPy, scikit-learn, OpenCV, ROS, LabVIEW, Arduino	