

Nithin Raghavan

(678) 200-5839 — rnithin@berkeley.edu — rnithin1 (Github) — csua.org/~rnithin

EDUCATION

University of California, Berkeley (2017 - 2021)

Aug 2017 – present

Computer Science *Bachelor of Arts*, Applied Mathematics *Bachelor of Arts*

(GPA: 3.67)

- CS61B: Data Structures
- CS170: Efficient Algorithms
- EE127: Optimization Models and Applications
- Math 128a: Numerical Analysis
- Math 126: Partial Differential Equations
- CS189: Introduction to Machine Learning

Georgia Institute of Technology

Aug 2015 – May 2017

Courses Taken in High School

- Math 3012: Applied Combinatorics
- Math 2803: Number Theory and Cryptography

EXPERIENCE

→ Visual Computing Lab, UC Berkeley

Oct 2019 – Present

- Worked with several graduate students to submit a paper to NeurIPS on a new concept in multilayer perceptron theory
- Theory states that an input embedding of Fourier Features enables a low-dimensional MLP to learn high frequency functions
- Helped research volumetric octree compression on a voxel grid for the Neural Radiance Functions (NeRF) paper
- Currently researching several concepts in graphics involving radiance transfer and volumetric rendering
- Researched NTK theory, neural network theory, kernel regression theory, measure theory, Fourier theory, signal processing, and relevant graphics knowledge for the paper

→ Ford Greenfield Labs

June 2020 – August 2020

- Worked on a neural network architecture to generate depth and segmentation maps from a single RGB image
- Reduces cost to almost zero, compared to several thousand dollars currently required to generate such real-world info
- Invention disclosure (that might result in a patent) submitted for consideration by Ford lawyers
- Currently writing a paper to be submitted to CVPR
- Additionally, worked on dynamic route calculation project for Ford Electric Vehicles taking into account charging stations, weather, cost and distance of travel

→ Samsung Advanced Computing Lab

May 2019 – August 2019

- Conducted extensive research on deep learning usecases and models as part of Samsung's GPU team
- Conducted extensive research on the potential routes of optimization and quantization of deep learning models such as MobileNet, R-FCN, SRCNN and ESRGAN as part of Samsung's GPU team
- Wrote and implemented OpenCL and OpenGL code
- Researched the graphics pipeline and became acquainted with AMD's compute and graphics architecture, and wrote + ran 2D register-blocked GEMM kernels with increased WPT in OpenCL on AMD architecture
- Wrote and trained two neural networks; the first to perform ambient occlusion on complex OpenGL-rendered scenes, and the second to convert a flat-rendered scene to a lifelike, physically based rendered one

→ Mobile Sensing Lab, UC Berkeley

Oct 2018 – Sept 2019

- Wrote code implementing a parallelized Frank-Wolfe algorithm for dynamic traffic assignment in C++/CUDA using contraction hierarchies
- Helped research the impact of different optimization models of routing behaviour on the Waze problem

→ RISE Lab, UC Berkeley

Jun 2018 – Dec 2018

- Designed and implemented a data visualization tool for Jupyter Notebook for hyperparameter optimization for Cirrus, a serverless machine learning framework
- Helped write code involving AWS Lambdas for model primitives such as logistic regression

→ IBM Almaden Research Center, Machine Learning Laboratory

Jul 2017 – Aug 2017

- Researched neural network architectures for the task of visual question answering on Stanford's CLEVR dataset
- Included LSTM sequence autoencoders in conjunction with CNNs

- **Georgia Tech School of Aerospace Engineering** *Sept 2016 – May 2017*
- Researched development of high bandwidth, high efficiency wireless energy transfer methods
 - Proposed circuits with millimeter wave input and Fabry-Perot resonators
- **Georgia Tech School of Physics** *May 2016 – Jul 2016*
- Shadowed professors and graduate students researching impacts of the September 2015 LIGO sighting of gravitational waves
 - Was introduced to the Einstein Toolkit for the modelling of relativistic astrophysical phenomena

PROJECTS

- **Resource-Provisioning GPU Server** *Dec 2017 – present*
- Developed a Python-based shell to automate on-demand request processing and resource provisioning in a GPU + CPU cluster
 - Collaborated on a team to create a program that utilizes Slurm for cluster management and deploys tasks in Docker containers
- **Software Renderer** *Jul 2019*
- Developed a software-based rasterizer and renderer with pixel and vertex shader support in C++
 - Capable of barycentric interpolation, backface culling and block-based rasterization
- **LASSO/Wavelet Based Compressed Sensing** *Jul 2019*
- Computes LASSO on the matrix-vector product representation of the discrete wavelet transform of an input signal with orthogonal Daubechies wavelets
 - Can lossily compress audio/images to any amount or preprocess them for ML training purposes
- **TaxiFindMe** *Apr 2018*
- Routing web app that helps New Yorkers find the best spot to minimize taxi waiting time, taking into account travel time and time of day
 - Preprocessed 20 million entry taxi dataset with k-means machine learning algorithm; for querying, KNN is run from an input location to find nearest cluster. Frontend employs Django
 - Reduced query time up to 94% from the naive implementation
- **ShirtMapper** *Jan 2018*
- App that resizes images of custom shirts and maps them onto people
 - Utilizes OpenCV and Scipy, and uses Haar classifiers for edge detection; frontend employs React Native

SKILLS

Awards: Exploravision National Contest *2016*

- Wrote a paper proposing blockchain's potential link to autonomous vehicles, and won honorable mention.

Models/Algorithms: Regression/classification (ridge, logistic, SVM, decision trees, OLS), PCA/SVD, ensemble learning, k-means, deep learning (CNNs, LSTMs, GANs), Frank-Wolfe

Frameworks/Softwares: Numpy, Scipy, Pytorch, OpenCV, Docker, Slurm, d3js, OpenCL, OpenGL

Programming Languages: Python, Java, C, C++, C#, Bash, Latex, SQL, JavaScript, Matlab

Operating Systems: Unix-like systems (Linux, FreeBSD, Mac OS X), Windows

Certifications: Android Development (University of Maryland through Coursera)