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

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 MLP theory: 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

→ **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 generate such maps to zero, compared to thousands of dollars currently required
- Invention disclosure (that might result in a patent) submitted for consideration by Ford lawyers
- Currently writing a paper to be submitted to CVPR

→ **Samsung Advanced Computing Lab** *May 2019 – August 2019*

- 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
- Researched the graphics pipeline and became acquainted with Samsung's future compute architecture, and wrote + ran 2D register-blocked GEMM kernels with increased WPT in OpenCL on Samsung 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

PROJECTS

→ **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

→ **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

→ **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

SKILLS

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